ECOLE NATIONALE
SUPERIEURE DE
BIBLIOTHECAIRES

DESS Informatique
Documentaire

RAPPORT DE STAGE

Conception et réalisation d'un système de prêt automatisé pour la bibliothèque de l'E.N.S.B.

Anne - Marie THUMMEL

Sous la direction de
Danielle ROGER
Conservateur de la bibliothèque
de l'E.N.S.B.



1990

Titre :Conception et réalisation d'un système de prêt automatisé pour la bibliothèque de l'E.N.S.B.

Auteur : Anne-marie THUMMEL

Stage effectué du 1er juin au 30 septembre 1990
à l'ECOLE NATIONALE SUPERIEURE DE BIBLIOTHECAIRES.

Directeur : Mr Jacques Keriguy

Adresse : 17/21 Bv du 11 novembre 1918
69523 VILLEURBANNE
Tel.: 78 89 64 45

Résumé : La bibliothèque de l'E.N.S.B. (Ecole nationale supérieure de bibliothécaires) a automatisé son service de prêt avec un programme "sur mesure" réalisé par un élève de l'Ecole. Ce programme a été écrit en DBASE IV Version 1.0.

Descripteurs : Prêt ; Automatisation ; Logiciel d'application DBASE IV.

Abstract : The library of the national librarian-school E.N.S.B. (Ecole nationale supérieure de bibliothécaires) has automated its loan department with an application software written in DBASE IV Version 1.0 by one of the school-students.

Keywords : Loan department ; Automation ; Application software DBASE IV.

[ ] Possibilité de diffusion [X] Confidentiel

TABLE DES MATIERES

I INTRODUCTION

L'objectif de notre stage était de réaliser un système de prêt automatisé pour la bibliothèque de l'Ecole Nationale Supérieure de Bibliothécaires (ENSB)
Cette école forme les cadres de bibliothèques ou services de documentation, notamment les conservateurs des bibliothèques d' l'Etat et des Collectivités publiques. Elle coordonne sur le plan national, la formation du personnel technique.

Devenue depuis 1990 Etablissement public à caractère scientifique culturel et professionnel (EPSCP), l'ENSB est habilitée à délivrer deux diplômes de 3e cycle correspondant à deux spécialisations : le DESS Direction de projets culturels et le DESS Informatique documentaire.
L'école participe en collaboration avec les universités de Grenoble et de Lyon, aux trois options des enseignements du DEA Rhône-Alpes "Sciences de l'information et de la communication".
Elle assure la formation de 80 à 200 étudiants par année universitaire.

II LA BIBLIOTHEQUE DE L'ENSB

1 Le public

Les utilisateurs habituels de la bibliothèque sont les étudiants de l'ENSB. S'y ajoutent les professeurs, le personnel de l'école, des étudiants de l'université de Lyon concernés par les sciences de l'information, du personnel des bibliothèques en formation continue, et un certain nombre de lecteurs "autorisés".
Si le nombre de lecteurs inscrits s'élève cette année à 250, le public potentiel est évalué à environ 500 inscrits. Le nombre d'étudiants est doublé à partir de la rentrée prochaine.

2 Le fonds

Les collections de son fonds sont par conséquent très spécialisées. Elles se rapportent avant tout au programme d'enseignement de l'école : description bibliographique, classification, bibliothéconomie, administration, gestion, marketing, nouvelles technologies appliquées aux sciences de l'information,...etc.
Le volume de ce fonds est :

Ouvrages	. Total : 20 à 25000 ouvrages En libre accès : évalué aux 3/4 du total
Périodiques	. Total : 376 titres En cours : 210 En libre accès : tous les titres vivants.

3 Le service de Prêt.

3.1 Activité

La bibliothèque est ouverte du lundi au samedi matin. Elle offre, comme nous le voyons, l'accès direct à une grande part de ses collections.
Les personnes pouvant participer aux permanences du service sont le personnel de la bibliothèque (2 bibliothécaires-adjoints, 1 magasinier (70%) ; et un bibliothécaire-adjoint d'un autre service de l'école, soit 4 personnes.
Les derniers chiffres indicatifs de l'activité du prêt sont les suivants, en moyenne des prêts à domicile / jour :

Année 89/90 :

Octobre	17
Novembre	27
Décembre	25
Janvier	23
Février	30
Mars	24
Avril	25
Mai	19
Juin	8
Juillet	4
Août	0 (fermeture)

Soit une moyenne annuelle de 20 prêts quotidiens.

Maximum de prêts mensuels en 1990 : 730 (février)
Moyenne des prêts en cours : 120 à 150

Ces chiffres ne sont guère impressionnants et ne rendent pas compte des services rendus par la bibliothèque pour les consultations sur place. Une activité fébrile s'y déploie, notamment à l'occasion des travaux pratiques demandés aux étudiants en recherches bibliographiques. Ces derniers sont nombreux aussi à consulter régulièrement les revues professionnelles et les ouvrages de base ne pouvant être empruntés à domicile.

3.2 Première informatisation

La bibliothèque de l'ENSB a informatisé une première fois le Prêt en avril 1985, pour deux raisons principales :

- Donner aux étudiants une notion concrète du fonctionnement d'un prêt automatisé tel qu'il se mettait en place alors dans les bibliothèques universitaires sous l'impulsion de la DBMIST. (Direction des bibliothèques des musées et de l'information scientifique et technique)

- Rendre les opérations du prêt plus simples et plus rigoureuses, le public de lecteurs étant appelé à augmenter.

Le logiciel mis en oeuvre était MOBI-PRET. Testé dans les bibliothèques à partir de 1982, ce système devint caduc en 1987. La société productrice céda ses droits à ISL qui commercialisa une version nettement améliorée, MOBI-BOP. Il devenait dès lors impossible de continuer la maintenance de l'ancien produit.

Il s'avérait que MOBI-PRET offrait peu de sécurité sur les données, en cas de micro-coupures par exemple. Au bout d'un certain temps apparaissaient des anomalies impossibles à corriger. Par ailleurs, les programmes de statistiques fournissaient des résultats assez limités.

Le prêt automatisé fut abandonné en mars 1987, le système manuel fut remis en place en attendant l'installation de MOBI-BOP. Mais le projet fut abandonné pour des raisons budgétaires. A la rentrée 1988, l'on remit en activité le système MOBI-PRET qui malheureusement tomba en panne en janvier 1989.

3.3 Fonctionnement actuel

Depuis janvier 1989, la gestion du prêt est redevenue manuelle à titre provisoire. Cette gestion repose sur le classement de fiches décrivant les documents susceptibles d'être empruntés à domicile. Une fiche comprend les informations suivantes :

- Cote
- Titre
- N°inventaire

En cas de prêt :

- N°du lecteur
- Date de retour prévue

Prêts

- Monographies

Livres non empruntés, leurs fiches sont classées dans un fichier "ouvrages" par ordre de N inventaire.

Livres empruntés, les fiches correspondantes quittent le fichier "ouvrages" pour prendre place dans un fichier "échéancier". Elles sont classées par ordre de date de retour. L'ouvrage est équipé d'un feuillet contenant les dates de retour successives.
Au retour la date est rayée sur la fiche et celle-ci reprend sa place dans le fichier "ouvrages".

- Périodiques

Un numéro de revue ne disposant pas de fiche, les prêts de revues sont consignés sur feuille libre. Celle-ci sera intercalée avec les autres prêts dans le fichier "échéancier".

Statistiques
A chaque "sortie" l'on appose un tampon "date du retour" sur une feuille journalière. 1 tampon = 1 prêt.

Réservations
Au retour d'un document, l'on place en réservation sur un rayon d'attente prévu à cet effet l'ouvrage dont la fiche comprend une note de réservation.
Un courrier est envoyé au lecteur concerné.

Relances
Les fiches des documents en prêt étant classées par ordre de date de retour, il est possible de repérer les retards et de compléter les formulaires de relance (le N de lecteur permet de retrouver son adresse dans le fichier "lecteurs"). Une marque est apposée sur la fiche des ouvrages concernés.

Courriers aux lecteurs
Ils sont relativement faciles à traiter. Le passage par les services postaux est rare car la majorité des lecteurs fréquentent régulièrement l'école et y possèdent un casier personnel.

3.4 Critique

Le volume d'emprunts à domicile est actuellement faible.Les opérations manuelles de prêt, même si elles sont un peu fastidieuses sont, selon nous acceptables du point de vue rapidité. Cependant un tel système présente les inconvénients suivants :

- **Prêt** : Le prêt de revues demande pour chaque fascicule, l'établissement d'une note provisoire qui devra être refaite lors d'un prêt ultérieur de ce même fascicule.

- **Statistiques** : les seules données statistiques fournies sont les nombres de prêts. Il n'est réalisé aucun tri sur le type de documents prêtés, le secteur documentaire ou les catégories de lecteurs. Ces données seraient d'une utilité évidente pour l'évaluation de l'efficacité et des besoins.

- **Réservations** : la gestion de plusieurs réservations sur un même document devient compliquée. De plus, un ouvrage réservé est souvent immobilisé trop longtemps, et le lecteur ayant réservé ne vient pas nécessairement emprunter le document.

- **Situation d'un document** : pour savoir qui a emprunté, et éventuellement réservé un document dont la fiche n'est pas dans le fichier "ouvrages", il faut parcourir l'ensemble du fichier "échéancier".
  Dans le cas d'un numéro de revue absent des rayons, même processus pour savoir si la revue est hors murs, et quel est l'emprunteur.
  Pour un volume de prêts important ce type de classement manuel serait totalement inadéquat.

- **Situation d'un lecteur** : elle ne peut être connue qu'en parcourant l'ensemble du fichier "échéancier" avec la contrainte d'avoir à repérer visuellement le N du lecteur sur chaque fiche.

- **Rigueur de gestion** : la fiabilité est relative étant donnés les risques d'erreurs d'annotations, d'oublis.
La gestion des autorisations et pénalisations est faite "au jugé". Elle dépend dans une certaine mesure de l'humeur et de la personnalité des employés au service du prêt. Nous tenons ici à remarquer que cette situation n'est pas forcément contestable lorsque la qualité des rapports avec le public est bonne. Elle peut parfois alléger les opérations. Toutefois, aux dires du personnel, elle est aussi une source de différends et de difficultés.

3.5 Nouveau projet d'automatisation du prêt

La régression vers un système de prêt manuel a été durement ressentie par le personnel de la bibliothèque. Elle est inacceptable pour une école de futurs cadres de services documentaires. Le service de documentation ne remplit pas sa mission en servant de modèle, non pas du présent ou du futur, mais du passé.

Pourquoi un système-maison ?

Les raisons qui s'opposent actuellement à l'acquisition d'un logiciel commercialisé tel que MOBI-BOP ou d'autres produits similaires sont :

a) **La faiblesse des moyens financiers** : l'acquisition de MOBI-BOP, par exemple, représenterait une dépense de l'ordre de 80 000 F.(le coût le plus important étant celui du matériel nécessaire). Cette dépense est incompatible avec le budget actuel de la bibliothèque.

b) **Manque de disponibilité du conservateur** pour procéder à une étude comparative des produits proposés par ailleurs.
Précisons que MOBI-BOP, indépendamment de son coût, est d'une puissance qui n'est pas nécessaire à un établissement de petite taille et de faible volume de prêt.

c) **Conditions locales privilégiées.**

- les étudiants préparant le DESS Informatique documentaire ont proposé, dans le cadre de leurs travaux pratiques, des réalisations simplifiées de systèmes de prêt. Ayant reçu une formation de base en programmation PASCAL et DBASE, un stage consistant à réaliser un système de prêt est l'opportunité d'approfondir leurs connaissances.

- si le programme est utilisable, il pourra à tout moment connaitre des adaptations et des développements, puisque étudiants et professeurs de l'école maitrisent le langage utilisé. Il est envisagé de résoudre de la même manière les problèmes de maintenance.

Il convient de conclure que cette option serait parfaitement erronnée dans une bibliothèque de type universitaire. MOBI-BOP a en effet prouvé son efficacité dans cet environnement. Les bibliothèques universitaires bénéficient d'ailleurs de budgets adaptés à ce type d'acquisition. Un logiciel maison poserait d'énormes problèmes de maintenance, d'autant que le flux annuel des transactions est de l'ordre de 45000 prêts pour une B.U. moyenne servant environ 5000 lecteurs.

III REALISATION DU PROGRAMME

1 Démarche de travail

1.1 Eléments de base

- **Cahier des charges.** (voir annexe 2)

Notre document de départ était un cahier des charges minimum réalisé par le conservateur de la bibliothèque. Nous nous y sommes référée sans pour autant y rester parfaitement fidèle pour la réalisation du programme. Des entretiens avec le conservateur de la bibliothèque nous ont conduite à modifier ou à préciser certains points.

- **Manuel d'utilisateur de MOBI-BOP**

Nous avons jugé intéressant de consulter le guide d'utilisation d'un système qui a fait ses preuves afin d'y puiser quelques idées pratiques de fonctionnement, de services offerts, d'organisation des fichiers.

- **Dialogue avec les futurs utilisateurs**

Nous avons demandé au personnel de la bibliothèdocumentation relative à l'ancien système MOBI-PRET. Nous souhaitions entendre aussi leurs opinions et souhaits pour le système à venir.

- **Démonstration** par son concepteur et programmeur, Mr AHA, du système automatisé de gestion d'un service documentaire de l'**INSA**. Ce programme a été écrit en DBASE III PLUS.

1.2 Progression

a) A partir des outils cités ci-dessus, nous avons dégagé les fonctions principales à mettre en place et schématisé l'essentiel du menu principal et des sous-menus.

b) Nous nous sommes ensuite représenté par une rédaction de type algorithmique la succession des tâches à effectuer. Ce travail nous a aidée à décider des fichiers à mettre en place et de leur structure (la nature et la forme des informations qu'ils devaient contenir) ainsi que des relations à établir entre eux.

c) Nous avons étudié le langage de programmation DBASE IV (voir annexe 1)
Le programme de notre enseignement comprenait l'initiation à DBASE III PLUS, langage de gestion de bases de données relationnelles qui est depuis 1984 un standard mondial. Cette initation nous a permis de nous adapter aux commandes et fonctions de DBASE IV. Ce langage est un produit DBASE de nouvelle génération fondé sur DBASE III PLUS.

En préalable à l'écriture du programme, nous avons consulté le guide Marabout "DBASE IV facile" (premier contact avec le langage), puis le Manuel de référence fourni avec le logiciel. Nous avons pris des notes détaillées et tâché de repérer quelles seraient les commandes les plus indispensables à l'écriture du programme.

d) Enfin, nous avons procédé à l'écriture proprement dite pour transformer en programmes les algorithmes de départ rédigés en langage naturel.
En cours d'écriture nous avons continué à découvrir la syntaxe des commandes et des fonctions. Ceci nous a amenée souvent à remanier l'écriture de modules réalisés précédemment. Les problèmes de syntaxe et un repérage "laborieux" des erreurs ont d'ailleurs considérablement ralenti notre progression en début d'écriture.

2 Principes retenus

2.1 Dans la conception du programme

Le but d'un tel programme est bien entendu de combler les lacunes et de remédier aux faiblesses de la gestion manuelle décrite dans la partie 3.3.

Les qualités essentielles que nous avons recherchées sont les suivantes :

- **Fiabilité** : gestion précise et rigoureuse des prêts et opérations annexes, sécurité des données.

- **Efficacité** : temps de réponse acceptables, automaticité maximale de certaines routines du prêt telles que
  . pénaliser les retards
  . supprimer les réservations après la date limite
  . calculer les dates de retour en fonction des fermetures
  . éditer après les relances la liste des lecteurs n'ayant pas répondu au 3e rappel.

- **Evolutivité** : structure modulaire facilitant des développements ultérieurs, effort de cohérence et d'organisation dans l'écriture.

- Ergonomie : simplicité et clarté de présentation, répondant autant que possible aux attentes des futurs utilisateurs.
Nous avons essayé
 . de disposer les écrans de façon homogène pour des contextes identiques
 . de disposer les messages d'erreur si possible toujours sur la même ligne d'écran, et de leur donner un aspect similaire.
 . d'éviter le bavardage (messages concis, limiter les questions à l'utilisateur)
 . de faciliter les corrections ou annulations d'opérations, de permettre des vérifications avant de valider.

- Souplesse : en particulier dans la gestion des dates de retour, des pénalisations et du nombre limite de prêts par usager. Mise en place de données paramétrables.

Ajoutons que nous avons voulu répondre aux besoins exprimés sous diverses formes par les futurs utilisateurs sans en rester totalement prisonnière, car il faut tenir compte des limites du langage de programmation...et des nôtres !

2.2 Dans l'écriture des programmes

- Indexation
 L'indexation permet des recherches plus rapides dans les fichiers.
 Nous avons utilisé la possibilité accrue d'indexation de DBASE IV. Le système peut mettre à jour jusqu'à 47 repères d'index dans un fichier unique d'extension .MDX (index multiple). Losque la base de données est modifiée, chaque repère d'index est mis à jour automatiquement. Il n'est donc pas nécessaire d'ouvrir de façon explicite le fichier index lors d'une mise à jour des données. 10 fichiers index peuvent être ouverts simultanément.

- Sécurité des données.
 Nous avons autant que possible manipulé les données à l'aide de variables. Ainsi les bases sont refermées avant le prochain dialogue avec l'utilisateur. Nous pensons ainsi diminuer les risques de détérioration des données suite à des manipulations erronées de l'utilisateur.

 Après des modifications sur les fichiers de données, les informations nécessaires à une éventuelle reconstitution sont imprimées automatiquement (journal).

 Nous avons utilisé dans le programme la fonctionnalité AUTOSAVE : Après une modification, DBASE IV effectue une mise à jour immédiate des fichiers sur le disque. Ceci évite des pertes de données dans le cas d'une coupure de secteur, ou autre incident.

 Un sous-programme de sauvegarde permet la copie sur disquette des fichiers de données et des fichiers index correspondants.

- Cohérence des fichiers
Suite à nos tests, nous avons introduit des contrôles systématiques de l'existence effective dans les fichiers, des codes ou autres valeurs entrées par l'utilisateur.

Nos tests avaient révélé par exemple que l'on pouvait enregistrer une réservation pour un lecteur inexistant.Nous souhaitons pallier à des cas de ce genre. Assez invraisemblables en pratique normale, ils peuvent se produire accidentellement..Il nous apparait préférable qu'un programme prévoie trop de fausses manoeuvres plutôt que trop peu.

- Identification des variables.
Les identificateurs commencent en règle générale par la lettre "V" (VINV=N inventaire ; VCAR=N de carte de lecteur...).
Nous n'avons pas multiplié les noms de variables si elles se rapportaient au même type de champ ou de valeur. Pour éviter les effets de bord, nous les avons déclarées selon les besoins, soit globales, soit locales.

- Passage de paramètres
Nous avons utilisé cette fonctionnalité de DBASE dans les procédures servant à un traitement répétitif sur des valeurs différentes. (programme des relances, recherches par mots du titre).

- Fenêtrage
Nous avons profité des commandes de fenêtrage pour
- l'affichage d'informations facultatives demandées par l'utilisateur
- obtenir des encadrés. Pour ce faire, le recours aux fenêtres n'est pas indispensable, mais il devient plus aisé de modifier la taille des encadrés qui servent dans plusieurs sous-programmes. Il suffit de changer dans le programme principal la définition de la fenêtre.

- Utilisation du générateur d'applications
Cette interface spécifique à DBASE IV nous a permis d'obtenir deux programmes de façon automatique. Nous avons ainsi créé des vues destinées l'une à l'édition de la liste des retards, l'autre à celle des prêts en cours. Nous avons ensuite recopié ces programmes en leur donnant l'extension .PRG. L'éxécution devrait en être plus rapide. Nous n'y avons rajouté que quelques lignes complémentaires.

Après quelques réalisations de menus déroulants (déplacement de pavés en surbrillance), nous les avons abandonnés afin de conserver l'homogénéité de nos écrans.
Nous avions ébauché la création de bases SQL. Finalement, nous nous sommes jugée trop inexpérimentée pour "mélanger" les deux modes dans un même programme.

Conclusion : Le temps et l'expérience faisant défaut, c'est avec mesure que nous avons exploité les nouvelles fonctionnalités de DBASE IV.

3 Structure du programme

3.1 Structure générale (voir aussi schéma page suivante)

C'est un programme modulaire arborescent dont le programme principal est BIBENS. Il appelle une série de 16 sous-programmes, certains de ces derniers appellent d'autres.

Certains programmes contiennent des procédures.
En effet, contrairement à DBASE III PLUS, DBASE IV accepte le concept de procédures à l'intérieur d'un programme. Ces procédures sont disponibles pour tout programme appelé ultérieurement.
Nous avons donc inclus dans le programme principal un certain nombre de procédures pour les routines devant être utilisées dans de nombreux sous-programmes. Au sein des sous-programmes, nous avons mis en procédures des traitements appelés à se répéter (messages d'erreurs, traitements communs à plusieurs choix du sous-menu).

Dans un souci de lisibilité nous n'avons pas fait figurer ces procédures dans le schéma (ci-après) de la structure d'ensemble.

En outre, pour simplifier la structure générale, nous avons écrit deux routines d'affichage sous forme de sous-programmes, et non de procédures.

(Le programme détaillé et commenté se trouve en annexe 3.)

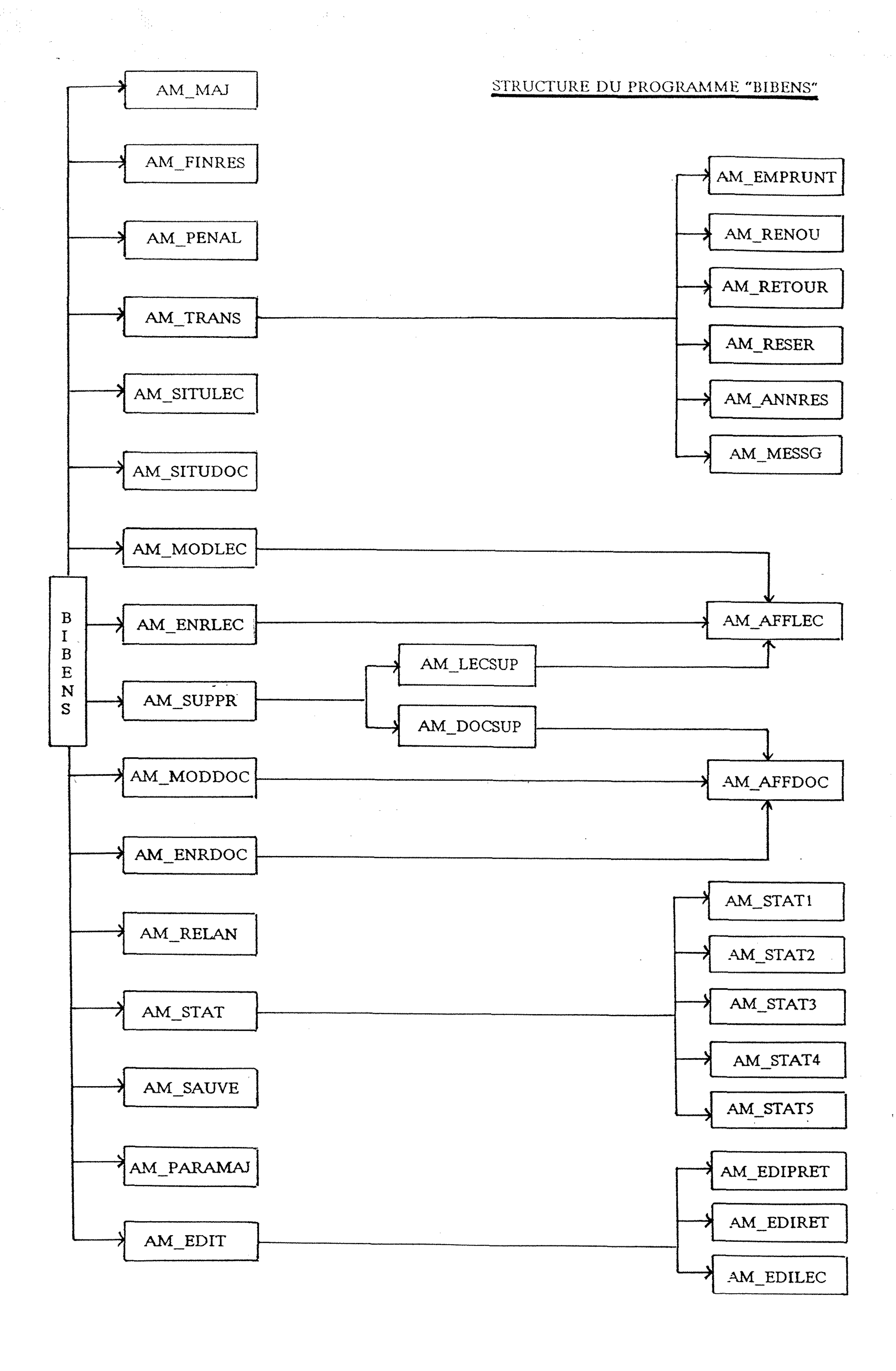
3.2 Les fichiers

Le programme comprend 8 fichiers de données (extension .DBF) et 4 fichiers d'index multiples (extension .MDX) correspondant respectivement à l'indexation de certains champs du fichier LECTEUR, DOCUMENT, PRET, COMPTEUR.

a)**Les fichiers ayant entre eux des relations :**
LECTEUR,DOCUMENT, PRET, RESERVAT

- Le fichier **LECTEUR**

Ce fichier contient les enregistrements relatifs aux coordonnées des lecteurs inscrits. Afin de faciliter la saisie, nous avons dissocié chaque unité d'information en autant de champs. Ex : NOM, PRENOM, RUE, CODE POSTAL...etc. L'enregistrement contient aussi le nombre de prêts et le nombre de réservations effectués par chaque lecteur. Un champ de pénalisation indique la date à partir de laquelle la pénalisation est levée. Une rubrique message est prévue également. Un champ indiquant la date limite de validité de l'inscription sera utile aux suppressions d'enregistrements des anciens lecteurs.


STRUCTURE DU PROGRAMME "BIBENS"
B I B E N S
AM_MAJ
AM_FINRES
AM_PENAL
AM_TRANS
AM_SITULEC
AM_SITUDOC
AM_MODLEC
AM_ENRLEC
AM_SUPPR
AM_MODDOC
AM_ENRDOC
AM_RELAN
AM_STAT
AM_SAUVE
AM_PARAMAJ
AM_EDIT
AM_EMPRUNT
AM_RENOU
AM_RETOUR
AM_RESER
AM_ANNRES
AM_MESSG
AM_AFFLEC
AM_LECSUP
AM_DOCSUP
AM_AFFDOC
AM_STAT1
AM_STAT2
AM_STAT3
AM_STAT4
AM_STAT5
AM_EDIPRET
AM_EDIRET
AM_EDILEC

- Le fichier **DOCUMENT**
L'enregistrement contient les coordonnées du document, une rubrique d'état des reservations, et une rubrique d'état de prêt. Les 5 champs suivants comptabilisent les prêts de l'année courante ainsi que les prêts des 4 années précédentes.
Nous répondons ainsi à une demande particulière du conservateur qui souhaite connaitre l'évolution des prêts des ouvrages.

- Le fichier **PRET**
C'est un fichier intermédiaire qui stocke durant un prêt de document le code lecteur, le code document, les dates, l'indication de renouvellement de prêt, et de relance du lecteur en cas de retard.
Renouvellement et relance sont des indicateurs permettant de bloquer, si nécessaire une tentative d'emprunt.

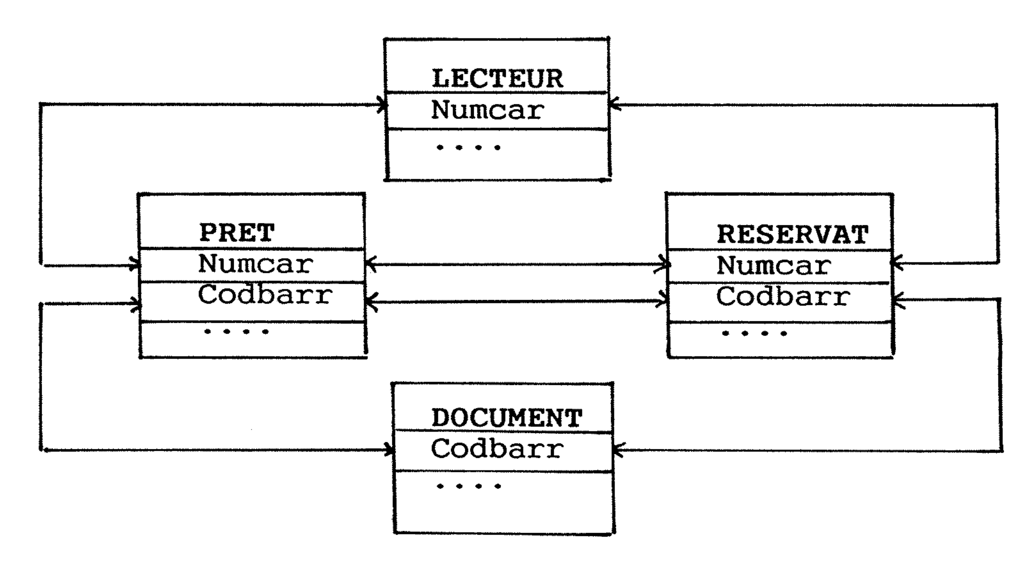
- Le fichier **RESERVAT**
C'est un fichier intermédiaire qui stocke durant le temps de réservation d'un document le code lecteur, le code document, la date limite de réservation.
Cette date limite permettra l'effacement automatique des réservations devenues caduques.

- **Relations entre ces fichiers**
Afin de limiter la redondance et donc la taille des fichiers, deux champs réalisent les liens nécessaires aux transactions de prêt ou de réservations et à la visualisation d'états lecteurs ou documents : NUMCAR et CODBARR.

NUMCAR désigne le code du lecteur . Il s'agit du N° de carte attribué automatiquement par le système lors de l'inscription du lecteur.

CODBARR désigne le code du document. Il s'agit de l'étiquette code à barre dont est équipé le document.
Ces deux codes sont des valeurs immuables qui identifient un seul lecteur ou un seul document tant qu'il reste enregistré dans la base du système.

Schéma des relations entre ces 4 fichiers :



b) **Les fichiers "indépendants" : PARAM, SECCAT, COMPTEUR, CATEG.**

Ces fichiers ne possèdent aucun champ de lien avec les autres fichiers.

- Le fichier **PARAM** contient un seul enregistrement. La valeur de ses champs est modifiable à tout moment excepté le champ MAJBASES. Ce dernier prend automatiquement pour valeur la date du jour à chaque mise en route du programme.
Les autres champs prennent des valeurs convenues pour le réglement du prêt : durées de prêt, intervalle entre les relances, dates de fermeture...Ces valeurs sont utilisées dans le programme principal par l'intermédiaire de variables globales servant à l'éxécution des sous-programmes.

- Le fichier **SECCAT**
SECCAT est utilisé pour l'édition de statistiques annuelles relatives aux secteurs documentaires et aux catégories de lecteurs.
Il contient autant d'enregistrements que de secteurs documentaires, soit 13. Les secteurs sont dans l'ordre de leur n d'enregistrement : P pour les périodiques ; A, B, C, AB, BA, BB, BC, CA, W, X, Y, Z, pour les ouvrages.

Chaque enregistrement contient 20 champs en prévision d'un nombre égal de catégories de lecteurs (Catégorie A à T en l'occurrence).

- Le fichier **COMPTEUR**
Il est utilisé pour éditer le nombre de prêts mensuels et annuels.
IL contient 13 champs. Le premier pour l'année considérée, les 12 autres correspondent aux 12 mois de l'année.

- Le fichier **CATEG**
Il comptabilise, par catégorie, le nombre de lecteurs inscrits.
Il contient donc 20 champs (20 catégories)

3.3 Commentaires

Les redondances d'information observées dans les fichiers LECTEUR (NBRESER, NBPRET) et DOCUMENT (RESER, PRET) sont destinées à éviter les ouvertures et lectures inutiles des fichiers PRET et RESERVAT.
D'une part, lorsque le lecteur ou le document n'est ni en situation de prêt ni en situation de réservation : les champs correspondants sont alors de valeur 0 et le programme peut continuer sans avoir à effectuer une lecture de contrôle dans les fichiers PRET et RESERVAT.
D'autre part, si la situation de prêt ou de réservation existe : la valeur des champs correspondants est alors <> 0.
Pour empêcher si nécessaire, une transaction ou pour obtenir un simple total, il sera inutile d'ouvrir et de lire PRET ou RESERVAT.

3.4 Structure des fichiers

```
. use compteur
. list stru
Structure de la base           : C:\DBASE\BIBENS\COMPTEUR.DBF
Nombre d'enregistrements       :        1
Date de dernière mise à jour : 16/09/90
Numéro Champ          Type      Largeur    Déc.     Index
     1  ANNEE          Numérique        4                O
     2  JAN            Numérique        4                N
     3  FEV            Numérique        4                N
     4  MAR            Numérique        4                N
     5  AVR            Numérique        4                N
     6  MAI            Numérique        4                N
     7  JUN            Numérique        4                N
     8  JUL            Numérique        4                N
     9  AOU            Numérique        4                N
    10  SEP            Numérique        4                N
    11  OCT            Numérique        4                N
    12  NOV            Numérique        4                N
    13  DEC            Numérique        4                N
** Total **                            53

. use seccat
. list stru
Structure de la base           : C:\DBASE\BIBENS\SECCAT.DBF
Nombre d'enregistrements       :       13
Date de dernière mise à jour : 16/09/90
Numéro Champ          Type      Largeur    Déc.     Index
     1  SECTEUR        Caractère        2                N
     2  CATA           Numérique        3                N
     3  CATB           Numérique        3                N
     4  CATC           Numérique        3                N
     5  CATD           Numérique        3                N
     6  CATE           Numérique        3                N
     7  CATF           Numérique        3                N
     8  CATG           Numérique        3                N
     9  CATH           Numérique        3                N
    10  CATI           Numérique        3                N
    11  CATJ           Numérique        3                N
    12  CATK           Numérique        3                N
    13  CATL           Numérique        3                N
    14  CATM           Numérique        3                N
    15  CATN           Numérique        3                N
    16  CATO           Numérique        3                N
    17  CATP           Numérique        3                N
    18  CATQ           Numérique        3                N
    19  CATR           Numérique        3                N
    20  CATS           Numérique        3                N
    21  CATT           Numérique        3                N
** Total **                            63
```

```
. use reservat
. list stru
Structure de la base         : C:\DBASE\BIBENS\RESERVAT.DBF
Nombre d'enregistrements     :      0
Date de dernière mise à jour : 04/09/90
Numéro Champ        Type      Largeur   Déc.    Index
    1  NUMCAR       Numérique       3               N
    2  CODBARR      Caractère       7               N
    3  DATELIM      Date            8               N
** Total **                        19

. use param
. list stru
Structure de la base         : C:\DBASE\BIBENS\PARAM.DBF
Nombre d'enregistrements     :      1
Date de dernière mise à jour : 16/09/90
Numéro Champ        Type      Largeur   Déc.    Index
    1  DUREE1       Numérique       2               N
    2  DUREE2       Numérique       2               N
    3  MAXPRET      Numérique       1               N
    4  DATEFERM     Date            8               N
    5  DATEOUV      Date            8               N
    6  INTERREL     Numérique       1               N
    7  DUREERES     Numérique       1               N
    8  MAJBASES     Date            8               N
    9  ACTIFS       Numérique       3               N
** Total **                        35

. use categ
. list stru
Structure de la base         : C:\DBASE\BIBENS\CATEG.DBF
Nombre d'enregistrements     :      1
Date de dernière mise à jour : 16/09/90
Numéro Champ        Type      Largeur   Déc.    Index
    1  CATA         Numérique       3               N
    2  CATB         Numérique       3               N
    3  CATC         Numérique       3               N
    4  CATD         Numérique       3               N
    5  CATE         Numérique       3               N
    6  CATF         Numérique       3               N
    7  CATG         Numérique       3               N
    8  CATH         Numérique       3               N
    9  CATI         Numérique       3               N
   10  CATJ         Numérique       3               N
   11  CATK         Numérique       3               N
   12  CATL         Numérique       3               N
   13  CATM         Numérique       3               N
   14  CATN         Numérique       3               N
   15  CATO         Numérique       3               N
   16  CATP         Numérique       3               N
   17  CATQ         Numérique       3               N
   18  CATR         Numérique       3               N
   19  CATS         Numérique       3               N
   20  CATT         Numérique       3               N
** Total **                        61
```

lecteur

```
. list stru
Structure de la base            : C:\DBASE\BIBENS\LECTEUR.DBF
Nombre d'enregistrements        :       3
Date de dernière mise à jour : 16/09/90
Numéro Champ         Type      Largeur   Déc.     Index
    1  NUMCAR        Numérique      3                 O
    2  CATEG         Caractère      1                 N
    3  NOM           Caractère     25                 N
    4  PRENOM        Caractère     25                 N
    5  RESID         Caractère     40                 N
    6  RUE           Caractère     40                 N
    7  CODPOST       Caractère      5                 N
    8  LIEU          Caractère     15                 N
    9  NBPRET        Numérique      1                 N
   10  NBRESER       Numérique      1                 N
   11  PENAL         Date           8                 N
   12  MESSG         Caractère    160                 N
   13  VALID         Date           8                 O
** Total **                       333

. use document

. list stru
Structure de la base            : C:\DBASE\BIBENS\DOCUMENT.DBF
Nombre d'enregistrements        :       2
Date de dernière mise à jour : 16/09/90
Numéro Champ         Type      Largeur   Déc.     Index
    1  TYPDOC        Numérique      1                 N
    2  CODBARR       Caractère      7                 O
    3  COTE          Caractère     10                 O
    4  NUMINV        Caractère      8                 O
    5  ANNEENO       Caractère     10                 N
    6  TITRE         Caractère     50                 N
    7  MOT1          Caractère     20                 N
    8  MOT2          Caractère     20                 N
    9  MOT3          Caractère     20                 N
   10  RESER         Numérique      1                 N
   11  PRET          Numérique      1                 N
   12  AN1           Numérique      3                 N
   13  AN2           Numérique      3                 N
   14  AN3           Numérique      3                 N
   15  AN4           Numérique      3                 N
   16  AN5           Numérique      3                 N
** Total **                       163


. use pret
. list stru
Structure de la base            : C:\DBASE\BIBENS\PRET.DBF
Nombre d'enregistrements        :       2
Date de dernière mise à jour : 16/09/90
Numéro Champ         Type      Largeur   Déc.     Index
    1  NUMCAR        Numérique      3                 O
    2  CODBARR       Caractère      7                 N
    3  DATESOR       Date           8                 N
    4  DATERET       Date           8                 N
    5  RENOU         Numérique      1                 N
    6  RELAN         Numérique      1                 N
** Total **                        29
```

IV GUIDE DE L'UTILISATEUR

BIENVENUE SUR "BIBENS"

BIBENS est un programme destiné à automatiser les principales tâches du Prêt à la bibliothèque de l'E.N.S.B.

Avant de prendre connaissance de ses divers services, lisez attentivement les explications suivantes. Elles vous fournissent les règles et principes généraux de fonctionnement du programme.

CONVENTIONS DE REDACTION DE CE GUIDE

<—┘ : symbolise la touche "Entrée"
>>> : ce que vous devez saisir ou répondre au système
* : réponses du système

I PRINCIPES DE FONCTIONNEMENT

1 Fonctionnement général

- **Journal imprimé** : Lorsque les opérations se déroulent normalement, elles sont suivies d'une impression . Le système prévoit l'impression des enregistrements, des modifications, des transactions de prêt, des éditions, des statistiques. Il n'imprime pas les visualisations.

En cas de panne, ce journal contient les données nécessaires à une reconstitution des opérations perdues.

- **Les erreurs.**

Si vous vous apercevez d'une erreur, il existe
. 2 possibilités d'annuler l'opération
. 1 possibilité de corriger.

ANNULER :- En début d'opération à la première ligne de saisie, tapez <—┘.
- En cours d'opération, quand il est impossible de circuler avec les touches "flèches", entrez des valeurs quelconques ou des blancs jusqu'à la question "Validation (O/N) ? ". Répondez N ou n.

CORRIGER : En cours de saisie, lorsqu'aucune rubrique n'est contrôlée, déplacez le curseur avec les touches "flèches", et corrigez.

- **Les champs obligatoires** (voir annexe): Pour l'enregistrement d'un document, certaines informations importantes doivent nécessairement être fournies. Si vous ne remplissez pas un champ obligatoire, le système vous le signale. Reportez vous à l'annexe pour mieux connaitre les rubriques à saisir pour les lecteurs et pour les documents.

- Conseils
Imprimante :
En début de journée, vérifiez que l'imprimante soit prête.
Si elle ne l'était pas, l'opération en cours est bien effectuée mais vous verrez un message du logiciel DBASE IV
"Imprimante non prête... Abandon- Ignorer - Suspendre."
Choisissez alors **Ignorer.**

- Sauvegardes :
N'oubliez pas de réaliser , avant de quitter le système, une sauvegarde sur 2 jeux de disquettes en alternance (un pour les jours pairs ; un pour les impairs)

2 Les écrans

- ...> Lorsque vous voyez apparaitre ce signe isolément ou à la suite d'un message, le système est en attente. Vous devez taper < pour continuer.

- (O/N) ? : Une question qui se termine ainsi accepte les lettres O ou o (pour Oui) ; N ou n (pour Non).

- **Messages** du système : Ils figurent généralement en bas d'écran.

- Conseils
. Lisez attentivement les messages ou questions du système.
. Ne faites pas de <—┘ intempestifs. Laissez se déplacer le petit tiret (-) sur l'écran tant que BIBENS ne vous propose pas de question ou n'affiche pas de message. .

II REGLES DE BASE FIXEES DANS BIBENS

Les secteurs documentaires.
Ils servent à l'édition de statistiques. Ces secteurs sont au nombre de 13 :

P : pour les périodiques
A,B,C,AB,BA,BB,BC,CA,W,X,Y,Z : pour les ouvrages. Ce sont les premières lettres de leurs cotes.

Les catégories de lecteurs.
BIBENS prévoit l'existence de 20 catégories allant de **A à T.**
Vous devez donc nommer vos différentes catégories de lecteurs par l'une de ces lettres.

Les types de documents
- Type 1 pour les documents dont la durée de prêt est la plus longue
- Type 2 pour les documents dont le prêt est plus court.

Les réservations.
- 3 réservations maximum sur un même document
- nombre de réservations illimité pour un lecteur en situation régulière

Les pénalisations
- Emprunt impossible en cas de pénalisation
  Remarque : vous pouvez provisoirement dépénaliser un lecteur en passant dans le choix "Modification lecteur".

Les renouvellements de prêts.
- Pour un document réservé, renouvellement impossible.
- Pour un lecteur en retard, renouvellement impossible.
- Pour un document de type 1, un seul renouvellement. Mais BIBENS vous donne le choix d'accorder malgré tout une prolongation, jusqu'à la limite de 3 renouvellements.
- Pour un document de type 2, 3 renouvellements possibles.

Les codes.
- Code lecteur : il s'agit du **N° automatiquement** attribué par le système lors de la première inscription d'un lecteur.
- Code document : il s'agit du **code à barre** inscrit sur l'étiquette dont **doivent** être équipés les documents destinés au prêt à domicile.

III REGLES QUE VOUS POUVEZ LIBREMENT MODIFIER

En demandant le choix (13)-"Paramètres" vous pouvez modifier :

- La durée de prêt des documents de type 1
- La durée de prêt des documents de type 2
- Le nombre maximum de prêts autorisés
- La durée de réservation
- Le nombre de jours d'intervalles entre deux relances.

Dans ces mêmes paramètres vous pouvez
- Entrer les dates de la prochaine fermeture (pour le calcul automatique de la date de retour des ouvrages).

IV LISTE DES FONCTIONNALITES

BIBENS vous propose deux menus importants qui sont :

- **MENU GENERAL** et **TRANSACTIONS** (voir page suivante)

A partir des choix effectués dans ces menus, certains modules proposent plusieurs mini-menus affichés dans une fenêtre.

Pour vous guider, nous vous proposons une liste de toutes les opérations réalisables par BIBENS, suivies du chemin à suivre pour y accéder. Les numéros sont les choix successifs à taper depuis le menu général jusqu'aux mini-menus.

Exemple : Liste des retards : (9)-(2)
(9) est le choix "Editions" du menu général
(2) est le choix "Liste des retards" du mini-menu des éditions.

* MENU GENERAL *

1 TRANSACTIONS	8 RELANCES
2 ENREGISTREMENT LECTEUR	9 EDITIONS
3 SITUATION LECTEUR	10 STATISTIQUES
4 MODIFICATION LECTEUR	11 SAUVEGARDES
5 ENREGISTREMENT DOCUMENT	12 SUPPRESSIONS
6 SITUATION DOCUMENT	13 PARAMETRES
7 MODIFICATION DOCUMENT	14 FIN DE JOURNEE

Entrez le numéro choisi >

Durée de prêt 1 : 14 Durée de prêt 2 : 7 Nb maxi de prêts : 3
Prochaine fermeture : 23/12/90 - 04/01/91

* TRANSACTIONS *

1 EMPRUNT

2 RENOUVELLEMENT

3 RETOUR

4 RESERVATION

5 ANNULATION DE RESERVATION

6 MESSAGE

7 RETOUR MENU GENERAL

Entrez le numéro choisi >

Durée de prêt 1 : 14 Durée de prêt 2 : 7 Nb maxi de prêts : 3
Prochaine fermeture : 23/12/90 - 04/01/91

Cette liste sera suivie d'explications plus précises indiquées dans l'ordre numérique des choix des menus

N.B.: La mention (auto) est relative aux opérations qui se réalisent automatiquement soit à l'allumage du système, soit après la fin d'une opération que vous demandez (Exemple : après les relances, BIBENS imprime la liste des 3emes relances qui n'ont pas eu de suite)

ANNULER UNE RESERVATION	(1)-(5)
DEPENALISER UN LECTEUR (=MODIFICATION)	(4)
EMPRUNT	(1)-(1)
ENREGISTRER UN DOCUMENT	(5)
ENREGISTRER UN LECTEUR	(2)
LISTE DES 3 EMES RELANCES SANS RESULTAT	(8)(auto)
LISTE DES LECTEURS NON-REINSCRITS	(9)-(3)
LISTE DES PRETS EN COURS	(9)-(1)
LISTE DES RESERVATIONS PERIMEES	(auto)
LISTE DES RETARDS	(9)-(2)
LISTE DES SUPPRESSIONS LECT. NON-REINSCRITS	(12)-(2)-(1)(auto)
MESSAGE POUR UN LECTEUR	(1)-(6)
MESSAGE POUR UN LECTEUR LORS D'UNE RELANCE	(8)
MISE A JOUR DES COMPTEURS	(auto)
MODIFIER LA DUREE DES PRETS	(13)
LA DUREE DES RESERVATIONS	(13)
MODIFIER L'ENREGISTREMENT D'UN DOCUMENT	(7)
En entrant par son code à barre	(7)-(2)
par sa cote	(7)-(3)
par son N inventaire	(7)-(1)
MODIFIER L'ENREGISTREMENT D'UN LECTEUR	(4)
En entrant par son code	(4)-(2)
par son nom	(4)-(1)
MODIFIER L'INTERVALLE DES RELANCES	(13)
MODIFIER LES DATES DE PROCHAINE FERMETURE	(13)
MODIFIER LE NOMBRE MAXIMUM DE PRETS	(13)
PENALISATION D'UN LECTEUR	(auto)
REINSCRIPTION D'UN LECTEUR	(4)
En entrant par son code	(4)-(2)
par son nom	(4)-(1)
RELANCES AUX LECTEURS EN RETARD	(8)
RENOUVELLEMENT D'UN PRET	(1)-(2)
RESERVATION D'UN DOCUMENT	(1)-(4)
RETOUR	(1)-(3)
SAUVEGARDE DES FICHIERS	(11)
SORTIE DU SYSTEME	(14)

SUPPRESSION D'UN ENREGISTREMENT DOCUMENT	(12)-(1)
SUPPRESSION D'UN ENREGISTREMENT LECTEUR (Entrée par son code)	(12)-(2)-(2)
SUPPRESSION DES LECTEURS NON-REINSCRITS	(12)-(2)-(1)
SUPPRESSION DES RESERVATIONS PERIMEES	(auto)
TOTAL DES DOCUMENTS ENREGISTRES	(10)-(5)
TOTAL DES LECTEURS ACTIFS	(10)-(5)
TOTAL DES LECTEURS ACTIFS par catégorie	(10)-(5)
TOTAL DES LECTEURS INSCRITS	(10)-(5)
TOTAL DES PRETS D'UNE ANNEE DONNEE	(10)-(3)
TOTAL DES PRETS D'UNE ANNEE DONNEE	
Pour un secteur et une catégorie donnés	(10)-(2)
Pour tous secteurs, toutes catégories	(10)-(4)
TOTAL DES PRETS EN COURS	(10)-(5)
TOTAL DES PRETS PAR MOIS	(10)-(3)
TOTAL DES PRETS POUR UN DOCUMENT PRECIS	(10)
Pour l'année en cours	(10)-(1)
Pour les 5 dernières années	(10)-(1)
VISUALISER LA SITUATION D'UN DOCUMENT	
En entrant par son code à barre	(6)-(2)
par sa cote	(6)-(3)
En entrant par son N inventaire	(6)-(1)
VISUALISER LA SITUATION D'UN LECTEUR	
En entrant par son code	(3)-(2)
En entrant par son nom	(3)-(1)
VISUALISER LE(S) DOCUMENT(S) CORRESPONDANTS	
En entrant 3 mots du titre	(6)-(4)

V DETAIL DES OPERATIONS

1 TRANSACTIONS

(1)-(1)Emprunt

>>>Entrez le code du lecteur

* Affichage des coordonnées et situation
Signalement de non validité d'inscription
Signalement des pénalisations
Signalement du nombre maximum de prêts,
mais possibilité d'accorder un prêt supplémentaire.

>>>Entrez le code du document

* Affichage des coordonnées
Signalement d'une réservation pour un autre lecteur
affichage de la date retour prévue
possibilité de modifier la date retour
>>>Validez <─┘ ; ou modifiez la date puis <─┘
* Affiche le nombre de prêts actualisé.

>>>Vous pouvez enchainer sur un autre emprunt

(1)-(2)Renouvellement d'un prêt

>>>Entrez le N° inventaire du document

* Affichage de coordonnées
>>>Confirmez si le document correspond au N° inventaire
* Signale si le document est réservé
Signale si le lecteur est en retard
Signale si le prêt a déjà été renouvelé,
mais possibilité d'accorder un autre renouvellement
affiche la nouvelle date de retour prévue
>>>Validez <─┘ ; ou modifiez la date puis <─┘

(1)-(3)Retour

>>>Entrez le code du document

* Affichage des coordonnées
En cas de retard, affichage de la date de retour dépassée et de la durée de pénalisation.
Si le document est réservé, affichage du nom, prénom du lecteur l'ayant réservé, date limite de la réservation
affiche le nombre de prêts restant

>>>Vous pouvez enchainer sur un autre retour

(1)-(4)Réservation

>>>Entrez le N° inventaire

* Affichage du titre
Signale si le document a déjà été réservé 3 fois
>>>Confirmez si le titre correspond bien
>>>Entrez le code lecteur

* Signale si ce lecteur a déjà fait la même réservation
Affichage des coordonnées du lecteur
Affichage du nombre de ses réservations actualisé.

(1)-(5)Annulation de réservation

>>>Entrez le N° inventaire

* Affichage du titre
>>>Confirmez si le titre correspond bien
>>>Entrez le code lecteur

* Affichage les coordonnées du lecteur
Affichage du nombre de ses réservations actualisé.

(1)-(6)Message pour un lecteur

>>>Entrez le code lecteur

* Affichage de ses coordonnées
Affichage du message existant déjà

>>>Entrez votre message ou modifiez le précédent

2 ENREGISTREMENT D'UN LECTEUR

* Affichage d'un nouveau N° de code lecteur
>>>Complétez toutes les rubriques du tableau. Vous pouvez avec les touches "flèches" vous déplacer pour corriger, tant que vous n'avez pas validé la dernière rubrique (=Observation)
* Affichage de la date limite de validité d'inscription.

Remarque : S'il existe des **homonymes** (même nom et prénom), le système vous demande s'il s'agit du même lecteur avant de continuer sur les rubriques suivantes ; ceci afin d'éviter les doubles inscriptions.

>>>Validez ou non puis <—┘

3 SITUATION D'UN LECTEUR

(3)-(1)Recherche par le nom

>>>Entrez le nom du lecteur

* Affichage d'une liste du ou des lecteur(s) de ce nom

>>>Confirmez en tapant le code correspondant

* Affichage des coordonnées du lecteur, ses prêts et ses réservations.
  Vous pourrez demander en option des précisions sur les documents concernés.

(3)-(2)Recherche par le code

>>>Entrez le code du lecteur

* Mêmes informations que (3)-(1)

4 MODIFICATION LECTEUR

(4)-(1)Entrée par le nom

>>>Même principe d'entrée que (3)-(1)

* Affichage de toutes les rubriques, sauf le code. Ce code ne doit jamais être modifié.

>>>Vous pouvez
- Dépénaliser le lecteur : en effaçant la date de pénalisation
- Réinscrire le lecteur : en changeant la date de validité.
- Modifier n'importe quelle rubrique et circuler avec les touches "flèches.

(4)-(2)Entrée par le code

>>>Même entrée que dans (3)-(2)

* Mêmes possibilités que (4)-(1)

5 ENREGISTREMENT D'UN DOCUMENT

>>>Entrez le code à barre

* Signale si ce titre est déjà enregistré pour éviter les doublons

>>>Entrez le type (=1 ou 2)
>>>Entrez la cote
>>>Entrez le N° inventaire (rubrique obligatoire)
NOTA: si ce N° comporte une "/" vous devez la placer **en 6e position** dans l'espace alloué.
>>>Entrez l'année de parution pour un ouvrage
l'année et le numéro de fascicule séparés par une virgule, pour une revue (rubrique obligatoire)
exemple : 1990,2344

>>>Entrez le titre (rubrique obligatoire)
>>>Entrez les 3 premiers mots significatifs du titre.

NOTA : Pour cette saisie, impossible, (sauf entre 2 rubriques non-obligatoires), de revenir en arrière pour corriger. En cas d'erreur :
- soit ne pas valider
- soit valider, puis passer immédiatement au choix (7)"Modification document" pour corriger.

>>>Validez ou non, puis <┘

6 SITUATION DOCUMENT

(6)-(1)Recherche par N° inventaire

>>>Entrez le N° inventaire

* Affichage de l'état :
 - réservé ou non, par qui
 - le prêt
 Possibilité de demander des précisions sur l'emprunteur.

(6)-(2)Recherche par code document

>>>Entrez le code

* Affichage identique à (6)-(1)

(6)-(3)Recherche par la cote

>>>Entrez la cote

* Affiche la liste des document de même cote
>>>Entrez le N° inventaire de celui qui correspond au titre souhaité

* Affichage identique à (6)-(1)

(6)-(4)Recherche par mots du-titre

>>>Entrez les mots du titre (3 maximum)

* Affichage du 1er document correspondant : N° inventaire,cote, titre

>>>Tapez <┘ pour obtenir le document suivant...etc.

Pas d'affichage de situation. Pour cela, retournez à l'entrée par cote ou par N° inventaire.

7 MODIFICATION DOCUMENT

(7)-(1)Entrée par le N°inventaire

>>>Entrez le N°inventaire

* Affichage de toutes les rubriques, sauf le code document qui ne doit changer en aucun cas

>>>Vous pouvez modifier n'importe quelle rubrique et circuler avec les touches "flèches".

(7)-(2)Entrée par le code document

>>>Entrez le code à barre

* Mêmes possibilités que (7)-(1)

(7)-(3)Entrée par la cote

>>>Entrez la cote

* Même liste que (6)-(3)
* Mêmes possibilités que (7)-(1)

8 RELANCES

* Le système affiche les coordonnées de chaque lecteur "relancé" et vous propose d'entrer un message pour ce lecteur.

>>>Modifiez son message puis <—┘ , sinon tapez <—┘ pour lancer l'impression de la lettre.

NOTA : Après l'édition de toutes les lettres de relance, BIBENS édite immédiatement la **liste des 3emes relances non abouties**.

9 EDITIONS

(9)-(1)Liste des prêts

* Impression des prêts en cours
  (cote, N°inventaire,titre,date retour,code document)

(9)-(2)Liste des retards

* Impression des retards en cours
  (cote,N°inventaire,nom,date retour,relances)

(9)-(3)Liste des lecteurs non-réinscrits

* Impression de la liste de ces lecteurs
  (code lecteur,catégorie,nom,prêts,date de fin de validité)

10 STATISTIQUES

(10)-(1)Prêts par document

>>>Entrez le code à barre

* Affichage et impression
  - du total des prêts de l'année en cours
  - du total des prêts des 4 années précédentes.

(10)-(2)Prêts par secteur/catégorie

>>>Entrez l'année souhaitée
>>>Entrez le secteur souhaité
>>>Entrez la catégorie souhaitée

* Affichage et impression du nombre des prêts correspondants.

(10)-(3)Total des prêts

>>>Entrez l'année souhaitée

* Affichage et impression d'un tableau
  - du total des prêts de l'année donnée
  - et du détail mensuel.

(10)-(4)Tableau par secteur/catégorie

>>>Entrez l'année souhaitée

* Affichage et impression d'un tableau pour chaque secteur
  - du total des prêts
  - du détail mensuel .

(10)-(5)Etat des documents et lecteurs

* Affichage et impression d'un tableau indiquant
  - Le nombre total de documents enregistrés
  - Le nombre total des prêts en cours
  - Le nombre total de lecteurs enregistrés
  - Le détail du nombre de prêts en cours pour chaque catégorie de lecteurs.
  - Le nombre de lecteurs actifs.

11 SAUVEGARDES

Avant d'appeler cette procédure, introduisez votre disquette dans le lecteur de l'ordinateur
Tapez ensuite le choix (11)

12 SUPPRESSIONS

(12)-(1)Suppression document

>>>Entrez le N°inventaire

* Signale si le document est encore en situation de prêt.
  Dans ce cas, la suppression est impossible.
  Affichage des coordonnées du document.

>>>Confirmez la suppression

Si le document faisait l'objet de réservation, celles-ci sont effacées.

(12)-(2)Suppression lecteur

(12)-(2)-(1)Suppr. par date de validité

Cette suppression concerne les lecteurs non-réinscrits depuis la date que vous entrerez.

>>>Entrez la date

* Le système supprime les lecteurs qui n'ont plus de prêts en cours, et supprime leurs réservations éventuelles.
  Il imprime la liste des lecteurs supprimés.

(12)-(2)-(2)Suppr. d'un lecteur

>>>Entrez son code

* Signale si le lecteur a des documents en prêt. Dans ce cas, la suppression est impossible.
  Affichage des coordonnées du lecteur

>>>Confirmez la suppression

Si le lecteur avait des réservations en cours, celles-ci sont effacées.

13 PARAMETRES

>>>Vous pouvez valider ou modifier les règles énoncées dans le chapitre III.

14 SORTIE DE BIBENS

Avant de quitter le programme, pensez à effectuer les sauvegardes. En cas d'oubli, effectuez les dès la prochaine mise en route de BIBENS.

ANNEXE

- **M(auto)** : vous serez automatiquement en majuscules.
- **ob** : champ obligatoire
- **c** : calculé par le système

Champs LECTEURS

	CHAMP	Largeur maxi	Type de caractères
c	CODE LECTEUR	3	chiffres
	CATEGORIE	1	lettre M(auto)
	NOM	25	lettres M(auto)
	PRENOM	25	lettres M(auto)
	RESIDENCE	40	lettres
	RUE	40	lettres
	CODE POSTAL	5	chiffres
	LOCALITE	15	lettres M(auto)
c	PENALISATION	8	date (/ /)
	OBSERVATIONS	160	lettres, chiffres
c	VALIDITE	8	date (/ /)

Champs DOCUMENT

	CHAMP	Largeur maxi	Type de caractères
	TYPE	1	chiffre
	CODE A BARRE	7	6 chiffres,1 lettre M(auto)
	COTE	10	chiffres, lettres M(auto)
ob	N INVENTAIRE	8	chiffres, 6e car :'/'
ob	ANNEE,N	10	chiffres, 5e car :','
ob	TITRE	50	lettres M(auto)
	1er MOT-TITRE	20	lettres M(auto)
	2e MOT-TITRE	20	lettres M(auto)
	3e MOT-TITRE	20	lettres M(auto)

BIBENS a été écrit en langage DBASE IV Version 1.01 par Anne-Marie THUMMEL à l'Ecole Nationale Supérieure de Bibliothécaires en été 1990.

VI CONCLUSION ET EVOLUTION

Nous pensons avoir donné à ce programme les capacités de base pour gérer le prêt de la bibliothèque de l'ENSB.
Après quelques mois d'expérience, la bibliothèque verra peut-être la nécessité de supprimer ou d'ajouter quelques fonctionnalités. Parmi celles que nous n'avons pas mises en place, nous citerons quelques possibilités d'améliorations :

- Entrée dans le système par un mot de passe . Pour ce faire, il faut utiliser la commande PROTECT (voir le manuel "Fonctions avancées" p.14-1)

- Etablir le journal non pas sur l'imprimante (inconvénient : bruit dans la salle de lecture), mais sur un fichier. Il suffira d'ajouter au début du programme principal une commande SET...(SET PRINTER TO FILE <nom du fichier>)

- Ajout d'autres compteurs en fonction des statistiques souhaitées. Nous avions envisagé la réalisation d'un fichier de type historique pour stocker les références les plus utiles aux statistiques de prêt ou de réservation. Cependant à partir d'une quantité dépassant 1000 enregistrements, nous ignorons si DBASE IV possède la puissance nécessaire aux parcours multiples d'un fichier.

- Le contrôle sur l'entrée des codes à barre est assez sommaire.En cas d'erreur, soit le système indique que le document n'est pas enregistré, soit il affiche un titre qui ne correspond pas au document voulu. Si la bibliothèque n'est pas équipée d'un crayon-lecteur, des erreurs risquent de se glisser
- lors de l'enregistrement d'un nouveau document.
- lors du prêt.
Il nous semble nécessaire de connaitre le mode de calcul de ces codes. L'on pourrait ainsi ainsi prévoir l'affichage automatique du 7e caractère (caractère de contrôle).

- Après des tests approfondis, une éventuelle compilation augmenterait la rapidité du programme.

"La programmation est l'art d'utiliser la puissance des ordinateurs pour déguiser en intelligence leur extrême stupidité" [1] . C'est ce que nous avons tâché de faire de notre mieux...

(1) MEYER,B. ; BAUDOIN, C.- Méthodes de programmation. Paris : Eyrolles, 1978.

BIBLIOGRAPHIE

OUVRAGE

[1] LAURENT, Jean-Pierre.- Initiation à l'analyse et à la programmation. Paris : Dunod, 1982. 99 p. (Dunod informatique phase formation). ISBN 2-04-015466-3

DOCUMENTATION FOURNIE AVEC LE LOGICIEL DBASE IV

[2] **De DBASE III à DBASE IV.** La commande électronique,1989. [pagination multiple]. ISBN 2-7390-0014-6.

[3] **Fonctions avancées.** La commande électronique, 1989. [pagination multiple]. ISBN 2-7390-0014-6.

[4] **Guide DBASE IV** . Ashton Tate Corporation, 1988. [Dépliant]

[5] **Guide de référence**. La commande électronique, 1989. [pagination multiple]. ISBN 2-7390-0014-6

[6] **Initiation à DBASE IV**. La commande électronique, 1989. [pagination multiple]. ISBN 2-7390-0014-6

[7] **Installation en réseau local**. La commande électronique, 1989. [pagination multiple]. ISBN 2-7390-0014-6.

[8] **Introduction à DBASE IV** : Version DOS. La commande électronique, 1989. 19+4 p. ISBN 2-7390-0014-6

[9] **Introduction au point de disponibilité**. La commande électronique, 1989. [pagination multiple]. ISBN 2-7390 -0014-6

[10] **Langage de référence**. Ashton Tate Corporation, 1988. [pagination multiple]. ISBN 2-7390-0014-6.

[11] **Prise en main de DBASE IV**. La commande électronique, 1988. [pagination multiple]. ISBN 2-7390-0020-0.

[12] **Utilisation des menus**. La commande électronique, 1989. [pagination multiple]. ISBN 2-7390-0014-6.

[13] **Utilisation du générateur d'applications de DBASE IV.** La commande électronique, 1988. [pagination multiple]. ISBN 2 -7390-0014-6.

ARTICLES DE PERIODIQUES

Au sujet de MOBI-PRET

[14] MALET, Françoise.- Mobi-Prêt à la bibliothèque de l'Université de Paris I. **Bulletin des bibliothèques de France**, 1984, vol 29, n 1, p.44-49.

[15] WABLE, Jean-Claude.- Mobi-Prêt à l'ENSB. **Inter-CDI**, Juillet-août 1986, p. 42-44.

Au sujet de DBASE IV Version 1.0

[16] COLIN, G.- Première prise en main de dBase IV. **Soft & Micro**, 1988, Octobre, n 45, p.138-139.

[17] JACOVELLA, B.- Ashton Tate lève le voile sur dBase IV. **Soft & Micro**, 1988, Mars , n 39, p.58-59.

Au sujet de DBASE IV Version 1.1

[18] LANGLET, Pascal .- DBase IV 1.1 enfin là ! . **01 Informatique**, 1990, Août, n 1123, p.5.

[19] PAOLIN, P.- DBase IV enfin opérationnel. **Soft & Micro**, 1990, Septembre, n 66, p.21-22.



GROS PLAN

LOGICIEL

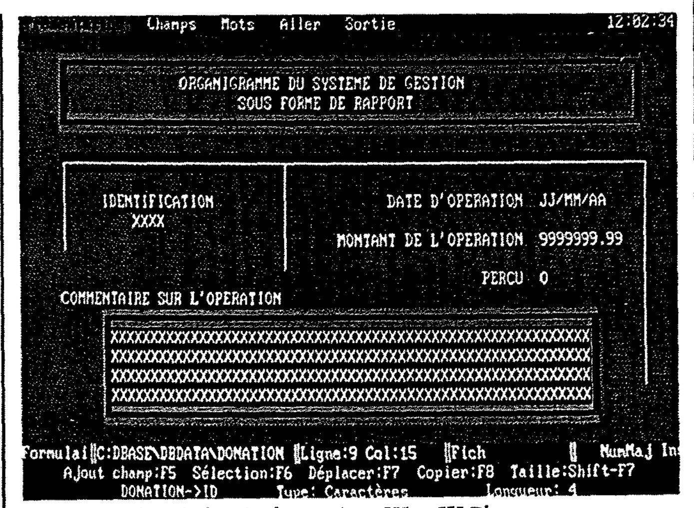
PREMIERE PRISE EN MAIN DE dBASE IV

Après son annonce prématurée et une présentation en fanfare cet hiver, les premiers bêta-tests de dBase IV (*) ont fait leur apparition. Il s'agit de versions MS-DOS partiellement francisées et livrées sans la documentation. Comme il peut y avoir des différences de vitesse et de fiabilité considérables entre ces pré-versions et le produit définitif, les évaluations chiffrées ne seraient pas significatives. En revanche, la structure générale du produit n'évoluera pratiquement plus, et les conclusions de ces bêta-tests ont toutes les chances de rester valables.

A la mise en route, une surprise attend l'utilisateur : dBase IV était présenté comme le continuateur de la série des dBase III et III Plus, mais on ne s'attendait pas à ce que les produits soient si proches. En effet, même si les progrès qu'apporte la version IV sautent aux yeux (l'interface utilisateur a été améliorée, la programmation est facilitée), sa structure reste fondamentalement celle des versions précédentes.

Les évolutions semblent surtout toucher « l'enrobage ». Ashton-Tate s'assure ainsi l'avantage d'une compatibilité presque totale avec un produit universellement répandu et qui représenterait quelque 70 % des ventes de SGBD en France. Il doit en contrepartie supporter beaucoup des limites de l'ancienne version. Ainsi, une des critiques les plus couramment formulées à l'encontre de dBase III est qu'il contraint trop rapidement l'utilisateur à programmer, en particulier pour développer des applications multifichiers interactives suffisamment conviviales et sûres. Le reproche reste valable pour la nouvelle version.

De la même façon, la procédure d'installation du produit est devenue plus conviviale (le programme



dBase IV : dans la lignée des versions III et III Plus.

d'installation comporte notamment un menu de sélection de pilotes d'impression), mais la moindre reconfiguration (le choix de nouvelles couleurs d'écran, par exemple) oblige toujours à modifier le CONFIG.DB à l'aide d'un éditeur de texte.

Comme dBase III Plus, dBase IV fonctionne selon trois modes : soit par commandes directes, soit par programme, soit au travers d'un ensemble de menus appelé bizarrement « centre de contrôle » qui correspond, dans son principe, au mode ASSIST des anciennes versions et qui, selon toute probabilité, sera le mode d'utilisation le plus fréquent.

En mode commandes, ce sont les progrès réalisés sur l'aide en ligne que l'utilisateur remarque : elle est plus fournie et « contextuelle » (les écrans d'aide changent selon la situation de l'opérateur). Une pression sur la touche de fonction F2 réaffiche l'écran du centre de contrôle.

■ Le centre de contrôle

Cet ensemble de menus déroulants fait largement appel à la notion de catalogue de fichiers apparue timidement sous dBase III Plus : tous les documents manipulés par une application (fichiers de données, écrans de saisie, rapports, vues...) sont référencés dans une table où l'utilisateur dispose d'une zone de saisie de commentaires. Les documents apparaissent à l'écran regroupés selon leur type, et un simple pointage par la touche Retour permet de les ouvrir (fichiers de données) ou de les mettre en service. Le centre de contrôle permet une manipulation beaucoup plus intuitive et plus rapide du produit, et il comporte des raffinements agréables, comme un utilitaire de gestion de disque (un DOS par menus déroulants) ou un système d'enregistrement et de modification de macrocommandes simple et agréable à utiliser.

Les fichiers de données sont créés à peu près de la même manière. Néanmoins, le nombre maximal de champs (zones de saisie) passe de 128 à 256 zones, et les données de type numérique « flottant » sont désormais possibles. Les champs mémo peuvent atteindre 64 Ko, et l'éditeur qui permet de les saisir est enfin de bonne qualité ; il offre des fonctions de recherche et remplacement, de mise en page et d'impression dignes d'un traitement de texte. C'est le même éditeur qui permet d'écrire et de modifier les programmes.

Les données en provenance de dBase III sont directement récupérables. De même, les index des versions précédentes restent utilisables, mais dBase IV propose de regrouper tous les index d'un fichier de données dans

(*) Les bêta-versions sont des sortes de prototypes au stade final.

un fichier d'index global. Ainsi, tous les index sont mis à jour simultanément, et l'opérateur passe d'un index à l'autre en indiquant simplement le nom de la clé à utiliser : le progrès est évident.

L'opérateur crée et appelle également ses formats de rapports, de saisie, d'étiquettes ou ses « vues » depuis le centre de contrôle, et il récupère sans problème ceux qu'il a défmis sous dBase III. Deux grosses différences de conception interdisent cependant de parler d'une compatibilité directe. La première est que, maintenant, les quatre modules (création de vues, d'étiquettes, de rapports ou de formats de définition d'écrans) génèrent tous un programme .PRG classique ; auparavant, ils produisaient autant de fichiers fermés et spécifiques. Le développeur acquiert une plus grande latitude d'intervention, puisqu'il peut modifier le code généré par dBase, puis le « recompiler ».

La seconde grande différence porte justement sur ce passage obligé par une pseudo-compilation des programmes. Cette façon de procéder, transparente pour l'utilisateur, est sans aucun doute rentable en ce qui concerne les étiquettes ou les formats de saisie (qui sont assez figés). Elle risque en revanche de pénaliser les utilisateurs qui modifient souvent leurs états d'édition.

Des quatre modules, c'est le générateur de rapports qui a le plus progressé. Il ressemble beaucoup à celui de Paradox 2. Comme lui, il propose des états en colonnes, en ligne (comme une facture par exemple) ou des lettres types. D'autre part, l'utilisateur crée à volonté des champs calculés, dispose d'une panoplie de fonctions statistiques et de larges possibilités de regroupement et de sous-totalisation des données. La présentation est soignée (lignes, cadres, graissage, soulignement), et l'impression est gérée par l'intermédiaire de pilotes évitant le recours aux codes de contrôle d'imprimante.

Le générateur de formats d'écrans est plus rigide qu'auparavant : il autorise des contrôles plus élaborés, mais reste monofichier lors des saisies. Le système d'édition d'étiquettes était déjà satisfaisant dans les versions précédentes. Il bénéficie principalement des progrès apportés à la gestion des périphériques d'impression.

■ Q.B.E. : la recherche par l'exemple

Un logiciel de gestion de données, même s'il est souvent utilisé à des traitements « lourds » de saisie ou d'édition, doit aussi être souple. L'information est souvent stockée dans différents fichiers qu'il faut d'abord mettre en relation grâce à des clés communes. Il faut aussi procéder à des sélections (prendre certains enregistrements et pas d'autres), aussi vite et facilement que possible. Après avoir introduit un système de vues sous dBase III, Ashton-Tate s'inspire cette fois de Paradox 2 et propose un système de Recherche par l'exemple (en anglais QBE, ou Query By Exemple).

L'utilisateur, par l'intermédiaire de menus déroulants, demande l'ouverture des fichiers dont il a besoin ; il désigne les colonnes de données qu'il veut visualiser, et lie les fichiers entre eux. Le tout s'effectue très simplement, pour peu que les index aient correctement été constitués et initialisés.

C'est lors des sélections que le bât blesse. D'abord, composer une condition de recherche n'est pas toujours simple ni intuitif. Ensuite, dBase recourt exclusivement à une exploration séquentielle des fichiers de données (il crée un « filtre »). Sur un fichier important, les temps d'attente risquent d'être très longs. De plus, la requête d'extraction se présente elle aussi sous la forme d'un programme qui doit impérativement passer par le pseudo-compilateur avant d'être exécuté. Attendons néanmoins la version définitive pour juger sans erreur possible de la rapidité du procédé.

■ Une programmation facilitée

Le mode programme est lui aussi accessible depuis le centre de contrôle, et il bénéficie d'un éditeur de texte enfin à la hauteur. Son langage de programmation a toujours été le point fort de dBase III, et ce sera en programmant que l'on appréciera le plus les nouvelles possibilités de dBase IV. En dehors d'une augmentation importante du nombre des fonctions (statistiques ou financières par exemple), il faut remarquer la multiplication de commandes d'interception réagissant à une erreur, une frappe clavier, un changement de page... et la gestion des tableaux à deux dimensions en mémoire vive. La présentation des applications sera améliorée par l'arrivée d'un mode de fenêtrage et d'un système de menus déroulants incorporables aux programmes. Comme avec la plupart des logiciels travaillant sous DOS en mode texte, la programmation de dBase IV n'est pas d'une facilité exemplaire dans ces deux derniers domaines.

Les micro-ordinateurs devenant de plus en plus fiables et puissants, les utilisateurs ont tendance à leur confier des informations de plus en plus sensibles. Pour répondre aux impératifs de sécurité, dBase IV dispose directement d'un système d'encryptage et de contrôle des accès à un fichier, et son fonctionnement fait appel à la notion de « transaction » des grosses bases de données SQL : quand cette fonctionnalité est activée, une série d'écritures sur disque n'est prise en compte que si elle a entièrement réussi. Dans le cas contraire, le système retourne à l'état antérieur : c'est le « rollback ». Cette possibilité, pas plus que la présence dans un sous-répertoire SQLHOME d'une sorte d'émulateur SQL, ne doit laisser croire que le logiciel d'Ashton-Tate est devenu l'égal d'Oracle ou Ingres : le SQL de dBase IV se contentera de traduire (à l'aide de tables d'équivalence) une série de commandes de syntaxe SQL en des commandes dBase telles que USE ou LIST.

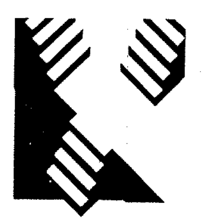
Enfin, le « centre de contrôle » de dBase IV incorpore un système de génération d'applications faisant un large usage des menus déroulants ; à première vue, toutefois, cela ne semble pas beaucoup plus performant que les différentes applications livrées avec les versions précédentes. Le module de génération est stocké dans un fichier extérieur à dBase ; il pourra sans doute être remplacé par des versions plus puissantes.

Prenant la suite d'un logiciel au succès commercial exceptionnel, dBase IV ne pouvait pas être révolutionnaire. Par rapport à son prédécesseur, il apporte principalement une interface utilisateur nettement plus agréable, une meilleure présentation des écrans et des sorties papier, et il incorpore des facilités de programmation que les développeurs devaient auparavant se procurer ailleurs.

Le « centre de contrôle », son mode de Recherche par l'exemple et ses macrocommandes permettent à un utilisateur débutant de maintenir des fichiers de taille modérée. Mais pour tirer de dBase IV ce que l'on est en droit d'attendre d'un gestionnaire de données de 8 450 F HT – à commencer par de vraies fonctionnalités de saisie interactive sur plusieurs fichiers –, il faudra se lancer dans la programmation : les outils offerts le permettent, et dBase IV confirme ainsi l'orientation comme outil de développement de ses prédécesseurs.

G. Colin

GUIDE DES MANUELS



Introduction à dBASE IV

Fascicule vous introduisant à dBASE IV™. Il comprend une liste récapitulative de la documentation, la configuration matériel et logiciel nécessaire ainsi qu'une présentation de chaque manuel. Il expose également les principes d'installation de dBASE IV mono-utilisateur. Vous devez lire ce fascicule en premier.



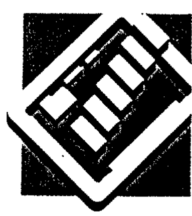
De dBASE III PLUS à dBASE IV

Fascicule offrant une vue d'ensemble des caractéristiques de dBASE IV. Il énumère notamment toutes les commandes et fonctions ajoutées ou modifiées par rapport à dBASE III PLUS™.



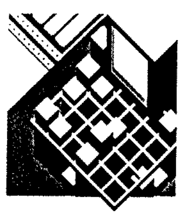
Guide de référence

Manuel de référence concis traitant des conventions, des commandes, des fonctions et des variables système dBASE®. Vous y trouverez également une brève description des commandes et des fonctions SQL, ainsi que des touches de déplacement du curseur.



Initiation à dBASE IV

Manuel permettant d'acquérir une expérience pratique du Centre de Contrôle de dBASE IV pour la création et la manipulation des bases de données. Il contient aussi une introduction à la méthode d'utilisation interactive de dBASE IV : le point de disponibilité.



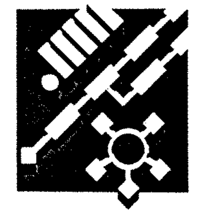
Utilisation des menus

Manuel offrant une vue d'ensemble du système de menus. Il comprend des instructions relatives à la création de bases de données, de catalogues, de vues, de rapports, de formulaires et d'étiquettes. Il explique comment modifier des données, imprimer des étiquettes et des rapports, et lancer des applications.



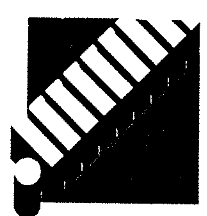
Introduction au point de disponibilité

Fascicule expliquant l'utilisation du point de disponibilité, interface de la ligne de commande dBASE. Vous acquerrez une expérience pratique en utilisant les fichiers exemple avec les commandes dBASE et apprendrez des techniques qui vous permettront de vous adapter aux fonctions avancées de dBASE IV.



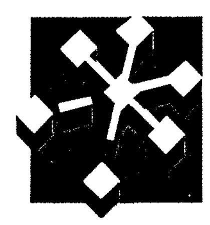
Fonctions avancées

Manuel vous introduisant aux fonctions les plus élaborées de dBASE IV. Il vous montre la programmation en langage dBASE, l'utilisation de SQL et celle de dBASE en environnement réseau local. Il n'est pas nécessaire d'être un expert dBASE pour lire ce manuel.



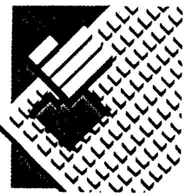
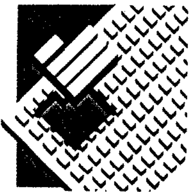
Langage de référence

Manuel de référence complet des commandes, des fonctions et des variables système du langage dBASE. Il présente également des exemples d'utilisation de ce puissant langage de base de données.



Installation en réseau local

Fascicule décrivant l'installation de la version multiposte de dBASE IV ainsi que la connexion d'utilisateurs au sein d'un réseau local. Il comprend également des informations spécifiques relatives à l'installation de dBASE IV sur plusieurs types de réseaux.

Utilisation du générateur d'applications de dBASE IV

Manuel exposant les principes de base de la conception d'applications ainsi que les méthodes permettant de créer rapidement des applications totalement fonctionnelles sans avoir recours à la programmation.

GUIDE DE L'ENVIRONNEMENT dBASE IV

Cette illustration vous montre comment les composants dBASE sont liés les uns aux autres et vous explique dans quels manuels rechercher des informations. Il vous sera parfois conseillé d'avoir recours au tutorial ou au système d'aide interactif.

Centre de Contrôle

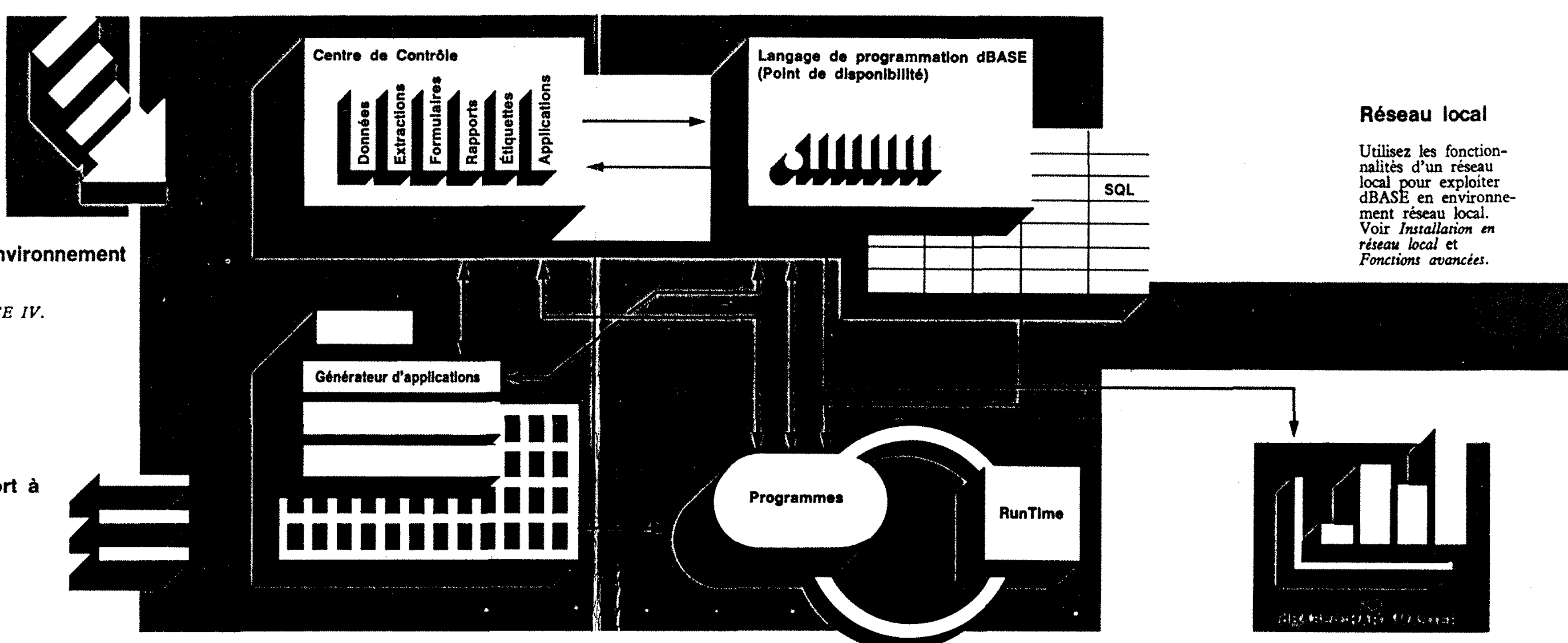
Utilisez le Centre de Contrôle pour pratiquer dBASE IV. Les utilisateurs néophytes comme les utilisateurs expérimentés peuvent facilement manipuler des fichiers et effectuer des tâches de gestion de bases de données, telles que la création et l'utilisation de bases de données, de catalogues, de vues, de formulaires, de rapports et d'étiquettes. Voir *Initiation à dBASE IV*, *Utilisation des menus*, l'aide en ligne et le tutorial.

Langage de programmation dBASE

Utilisez les commandes et les fonctions au niveau du point de disponibilité ou depuis les programmes pour accéder directement aux données et concevoir des applications dBASE. Vous avez également la possibilité de modifier des programmes créés avec le générateur d'applications. Voir *Introduction au point de disponibilité*, *Fonctions avancées*, *Langage de référence*, *Guide de référence* et l'aide en ligne.

SQL

Utilisez SQL pour combiner la norme de langage des grands systèmes avec la puissance et la souplesse du langage dBASE. Les commandes et les fonctions SQL permettent d'accéder aux données de façon interactive ou depuis des applications dBASE. Voir *Fonctions avancées* et l'aide en ligne.



Réseau local

Utilisez les fonctionnalités d'un réseau local pour exploiter dBASE en environnement réseau local. Voir *Installation en réseau local* et *Fonctions avancées*.

Découverte de l'environnement dBASE IV

Voir *Introduction à dBASE IV*.

Améliorations par rapport à dBASE III PLUS

Voir *De dBASE III PLUS à dBASE IV*.

Générateur d'applications

Utilisez le générateur d'applications pour créer des applications sans recourir à la programmation. L'apprentissage des bases de la conception d'applications est recommandé aux développeurs débutants. Les développeurs expérimentés peuvent gagner du temps, tout en conservant la possibilité d'utiliser le langage dBASE. Voir *Utilisation du générateur d'applications de dBASE IV* et l'aide en ligne.

RunTime *

Utilisez RunTime pour la diffusion de vos applications. Les développeurs professionnels peuvent, avec RunTime, distribuer des applications à leurs clients.

Ce composant n'est disponible qu'avec *dBASE IV développeur*.

Lien dBASE/CHART-MASTER

Utilisez cette interface entre dBASE et CHART-MASTER® pour concevoir et générer des graphiques de présentation professionnels à partir de vos données. Voir *Utilisation des menus*.

Bases de données

dBase IV 1.1 enfin là!

leilleure gestion de la mémoire, accélération es traitements, la version 1.1 de dBase IV, ujourd'hui disponible, suffira-t-elle à enrayer les ertes de marché auxquelles se trouve confrontée .shton Tate ?

Résultat d'un vaste pro-amme de tests, la version 1.1 de Base IV serait totalement débo-ıée. On se souvient encore de arrivée tardive et vivement criti-uée, en octobre 1988, de la der-ière version du numéro un des ases de données pour micros. Décidés à éviter les mêmes erreurs, :s dirigeants d'Ashton Tate se sont ıncés dans un large programme de :sts sur lesquels ont planché plus e cent trente personnes au siège méricain de l'éditeur et auquel se ont associés deux mille cinq cents ites de bêta-tests. Cinq millions de :sts représentant quarante-cinq ıille heures de travail, soixante ıille procédures de tests utomatiques: autant de chiffres ui donnent une idée des moyens ıis en œuvre par Ashton Tate, onsciente d'avoir *«commis cer-aines erreurs lors du lancement de lBase IV et décidée à ne laisser ortir la version 1.1 que lorsqu'elle ıurait été suffisamment testée our obtenir une version irrépro-:hable».*

:ontre le piratage, ın numéro encrypté qui dentifie l'utilisateur

Parmi les nouvelles fonctionna-ités de dBase IV 1.1, commerciali-é sans protection physique et avec ıne installation simplifiée, figure :ssentiellement une meilleure ges-:ion de la mémoire. L'installation du logiciel s'avère plus intuitive et guidée par des écrans d'aide. L'ab-sence de protection est compensée par l'encryptage d'un numéro de série qui permet de connaître aus-sitôt l'identité de l'utilisateur auto-risé. Rappelons qu'Ashton Tate et son distributeur français, La Com-mande Électronique, se trouvent à la tête des actions juridiques contre le piratage informatique.

Les fichiers système de dBase IV version 1.1 (dBase.EXE et dBa-se.OVL) ne requièrent plus que 410 Ko de RAM pour fonctionner. Par rapport à la version précé-dente, dBase IV 1.1 n'utilise que les ressources nécessaires à l'exécu-tion d'une commande. La version 1.0 nécessitait 480 Ko de RAM. La zone mémoire utilisée pour la ges-tion interne de dBase IV 1.1 se limite aujourd'hui à 330 Ko. Une zone tampon de mémoire interne de 80 Ko supplémentaires permet de conserver les segments d'«over-lay» (dépassements de capacité) les plus utilisés. Le gain en mé-moire s'élève donc à 70 Ko. C'est cette mémoire supplémentaire qui améliore les performances du logi-ciel.

La technologie DMMS (Dyna-mic Memory Management Sys-tem), à laquelle dBase IV doit ces performances, ne charge en mé-moire que les modules indispensa-bles. Elle permet de supprimer les restrictions liées aux UDF (User Defined Commands) et aux com-mandes ON (On Error, On Es-cape...). L'insertion d'un cache-dis-que en mémoire étendue ou paginée est destinée à *«augmenter de jusqu'à 400 % la vitesse de trai-tement de dBase IV».* Permettant de stocker de manière temporaire les fichiers et les données fréquem-ment utilisés, elle constitue la se-conde innovation majeure de la version 1.1. La plus impression-nante d'entre elles demeurant tou-tefois la faculté de paramétrer la mémoire RAM employée. L'utili-sateur attribue ainsi une zone de mémoire flottante à chaque mo-dule en fonction du type de travail effectué. Le gain en espace mé-moire peut alors atteindre 110 Ko.

Parmi les principales caractéris-tiques de la version 1.1 figurent encore les options du sous-menu «Organiser», intégré directement au niveau des écrans «Browse et Edit», qui permettent à l'utilisa-teur d'accéder directement aux fonctions d'indexation et de tri sans quitter pour autant le mode de présentation des données. En per-mettant la modification de la lar-geur des colonnes de présentation, le QBE (Query By Exemple) per-met de visualiser immédiatement les informations relatives à l'ex-traction en cours. Quant aux com-mandes SQL (Standard Query Language), elles sont de plus en plus nombreuses à être accessibles de façon transparente. Enfin, La Commande Électronique annonce la disponibilité d'un pilote d'im-primantes Postscript développé par ses propres soins.

La version réseau commercialisée fin août

Ces nouvelles fonctionnalités suffiront-elles à empêcher la pro-gression de Paradox (Borland), Foxbase (Fox Software), R:Base (Microrim) ou encore Dataease (Dataease), principaux concur-rents de dBase? Annoncé cette semaine à Londres, dBase IV ver-sion 1.1 sera disponible début août en version française. La Com-mande Électronique réalise encore près de 50 % des ventes sur le créneau des bases de données avec dBase III Plus qui semble parfois mieux correspondre aux besoins des utilisateurs disposant d'une plate-forme matérielle plus limi-tée. *«Nombre d'applications exis-tent et fonctionnent très bien sur dBase III Plus»*, déclare-t-on chez l'éditeur français. *«Pour la plupart de ces utilisateurs, la rapidité n'est pas un facteur motivant (du moins pour le moment) la mise à jour des programmes dBase III Plus.»*

Les utilisateurs de la version 1.0 pourront l'échanger contre la ver-sion 1.1 moyennant la somme de 150 FF (ht). Ceux qui utilisent toujours dBase III Plus devront débourser 2 500 FF (ht). Une ver-sion réseau de dBase IV sera en outre commercialisée fin août. Les développeurs disposeront d'une version spécifique à partir de sep-tembre. Quant à la version 1.1 Édition étendue, elle ne sera pas commercialisée avant 1991.

Pascal Langlet

JEAN-LOUIS MARX, ARTHUR ANDERSEN

«Dans l'immédiat, je ne vois pas l'intérêt d'abandonner dBase III»

Pour choisir une base de données micro, il faut bien cer-ner la nature de l'application. S'agit-il d'un SGBD de produc-tion ou d'une petite applica-tion? Pour le premier, selon Jean-Louis Marx, consultant chez Arthur Andersen, *«le point essentiel reste d'interfa-cer les bases de données SQL avec les applications exis-tantes sur les moyens ou les grands systèmes».*

Pour les petites applications, dit-il, *«nous n'utilisons pas dBase IV, mais le compilateur Clipper pour développer le pro-duit final».* En outre, *«dBase III Plus s'avère moins lourd pour créer une structure de fichier, moins gourmand en mémoire et en espace disque».*

«L'intérêt de dBase IV, pour-suit Jean-Louis Marx, *pourrait être d'offrir une interface avec SQL Server et Oracle. L'inter-face SQL n'étant pas encore disponible, quel intérêt vérita-ble y a-t-il à payer une mise à jour du produit (2 500 FF ht) pour passer de dBase III Plus à dBase IV?»* Certes *«le pas-sage des clients vers un mo-teur SQL est progressif, mais il devient de plus en plus d'actua-lité pour des applications moyennes. Le problème chan-ge avec le nombre d'utilisa-teurs. S'il se limite à deux ou trois, je ne vois aucun inconvé-nient à choisir dBase, mais s'il s'agit d'une centaine de postes à installer, le problème n'est plus le même. Le compilateur Clipper a une certaine cote auprès des entreprises parce qu'il n'impose pas de payer un nombre élevé de licences mais se limite uniquement au coût du développement.»*

Quant à savoir s'il y a d'au-tres produits intéressants, *«Paradox est un exemple, mais aussi SQL Windows qui devrait être disponible prochai-nement en interface avec Ora-cle. dBase IV a une longueur de retard sur Object View de Ma-tesys qui permet de dévelop-per des applications interfa-cées avec SQL Server. Win-dows et OS/2 Presentation Manager sont de plus en plus fréquemment choisis par les clients pour leur interface gra-phique. Il serait intéressant d'avoir un dBase IV sous Win-dows ou OS/2 PM.*

Portatifs

Sharp passe sous les deux kilos

Avec le PC 6220, un 286 VGA avec disque dur de 20 Mo, le constructeur japonais fait une entrée remarquée sur le marché très convoité des porta-tifs A 4.

Après Compaq et Toshiba, c'est donc au tour de Sharp d'an-noncer un micro-ordinateur au format A 4 (il mesure 21,6 × 27,9 cm) équipé d'un processeur 80C286. Zenith Data System (groupe Bull), acteur majeur du marché, *vient pour sa part* de lan-cer une version disque dur du Minisport, un 80C88 de 2,7 kg, en attendant d'introduire de nou-veaux modèles plus puissants. Phi-lips s'apprête également à mettre sur le marché européen une série de portatifs A 4 entrant dans cette catégorie. Mais, côté poids, la palme revient au Sharp PC 6220 qui, plus léger que chacun de ses concurrents équipés du processeur 80C286, pèse moins de 2 kg, contre 3,6 kg pour le T1 200XE de Toshi-ba ou le PCL 200 VGA de Philips et 2,8 kg pour le LTE/286 de Compaq.

Pour atteindre de telles perfor-mances, Sharp s'est débarrassé du lecteur de disquettes, optionnel et externe, et l'a remplacé par un disque dur de 2,5 pouces de 20 Mo, dont la taille n'excède pas celle d'un paquet de cigarettes (avec un temps d'accès moyen de 23 ms). La question est aujourd'hui de savoir quel choix technique s'imposera demain. Faut-il offrir à l'utilisateur un lecteur de disquettes interne qui grève le poids de la machine de quelques centaines de grammes et augmente son épaisseur de près d'un centimètre, ou bien le propo-ser en option dans un boîtier ex-terne?

Cette solution, retenue par Sharp, a déjà été proposée par NEC avec l'Ultralite (moins de 2 kg), commercialisé uniquement aux États-Unis, par Psion avec le MC 600 ou encore Poqet avec son PC 80C88 (avec 512 Ko de RAM et 640 Ko de ROM), disponible depuis peu en France. Ce dernier ne pèse que 450 grammes, mesure 22 × 10 × 2,5 cm et offre, en outre, une centaine d'heures d'autono-mie avec deux piles alcalines.

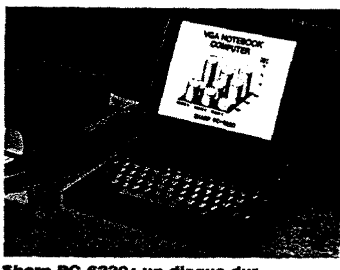
Sans complexe, la solution rete-nue par Sharp consiste à intégrer dans la mémoire ROM (non vola-tile) de 1 Mo la version 4.01 de MS-DOS, un logiciel de diagnos-tic, et Laplink 2.1, destiné au trans-fert des fichiers issus et repartant vers tous les ordinateurs équipés d'une sortie série RS 232C. La mémoire RAM de 1 Mo est, quant à elle, extensible à 3 Mo par incré-ment de 1 Mo. Le minimum requis pour faire tourner Windows 3.0 et la future version d'OS/2 est donc atteint.

Côté extension, le Notebook de Sharp dispose d'un port parallèle standard externe, relié par un mi-niconnecteur, et d'un port série RS 232C interne. L'emplacement d'une carte modem est prévu à l'intérieur du portatif, tout comme l'adaptateur optionnel pour un écran VGA externe.

L'un des points forts de ce nou-veau portatif A 4 286 est qu'il peut émuler le mode VGA (640 × 480 points) et afficher seize niveaux de gris sur un écran LCD page blanche triple supertwist (offrant un angle de vue plus large que les modèles double supertwist exis-tants). La diagonale de l'écran, rétroéclairé par un système fluores-cent à cathode froide, est de 25 cm.

Le prix du Sharp PC 6220, de l'ordre de 28 000 FF (ht), est de 10 % plus élevé que celui du Com-paq LTE/286 et est légèrement inférieur à celui du T1 200XE de Toshiba. Son épaisseur (3,4 cm) contre 4,8 cm et 4,4 cm pour Com-paq et Toshiba lui confère un avan-tage indéniable sur ces modèles. Pour mémoire, le Minisport de Zenith (commercialisé 17 000 FF ht) ne mesure que 3,3 cm de haut mais la diagonale de son écran CGA n'excède pas 9,5 pouces, soit une surface utile de 21 × 8 cm contre 20 × 15 cm pour le Sharp PC 6220. Un confort visuel parti-culièrement appréciable qui per-met au PC 6220, disponible en quantité à partir de septembre, d'offrir à ce jour le meilleur rap-port *prix/performances.* Si l'utili-sateur n'est pas trop attaché à son lecteur de disquettes interne...

PLa



Sharp PC 6220: un disque dur de 2,5 pouces de 20 Mo.

p-5

Système de prêt pour la Bibliothèque de l'ENSB :

La Bibliothèque compte 2 à 300 lecteurs ; une extrapolation sur 500 lecteurs semble suffisante.
La Bibliothèque compte 20000 ouvrages environ ; une partie du fonds étant exclu du prêt, on peut tabler sur un maximum de 10000 ouvrages susceptibles d'être prêtés. Toutefois, le prêt des fascicules de périodiques, certes occasionnel et faible, peut faire augmenter ce chiffre.

En raison de l'urgence, il est envisageable de mettre en route une version simplifiée d'un logiciel de prêt. Cette version devra être choisie en fonction de plusieurs critères, les deux étant impératifs :
- possibilité de développement de cette version pour satisfaire au cahier des charges complet sans avoir à reprendre la saisie des données.
- fiabilité
- ergonomie
-efficacité de la recherche ; temps de réponse courts
- facilité de mise à jour des données

Dans ce qui suit, on distinguera les fonctions à réaliser immédiatement de celles qui pourront l'être ultérieurement. Les secondes seront mentionnées précédées de ** (c'est le cas par exemple de tout ce qui concerne les réservations).

Conditions de prêt :
14 jours ou 7 jours selon les ouvrages.
Pour le personnel de l'Ecole, le prêt est autorisé pour un mois.
Le prêt est renouvelable une fois, sauf pour les livres prêtés 7 jours.

Fichier des documents :
Ce fichier devrait contenir au minimum :
Données concernant le document lui-même :
- la cote du livre (sur 10 caractères maximum).
- son numéro d'inventaire (sur 6 caractères)
- les 50 premiers caractères du titre
- 3 mots significatifs du titre

Dans le cas d'un périodique, le numéro d'inventaire peut être remplacé par le numéro de volume, suivi de celui de fascicule.

Pour ce fichier, la clé ne doit être aucun de ces éléments, car il n'y a pas unicité (le numéro d'inventaire d'un périodique, même si on lui ajoute l'année, ne détermine pas un exemplaire unique, à cause des fascicules).

Fonctions :
Il faut pouvoir rechercher un document :
- par son numéro d'enregistrement (clé)
- par sa cote
- par son numéro d'inventaire
**- par trois mots significatifs du titre

Le résultat de la recherche doit être l'affichage de la situation du document qui doit comporter :
- l'état du document : présent ; en prêt et à réserver ; en

prêt et sans réservation
- le nom et le numéro d'enregistrement de l'emprunteur actuel
- la date de prêt et la date de retour prévue
**- le nom et le numéro d'enregistrement du lecteur ayant réservé le document

Lorsque la recherche aboutit à plusieurs documents (exemple : recherche par cote), une étape supplémentaire doit permettre de sélectionner le document utile après affichage des différents documents proposés dans un format minimum : cote-inventaire-30 premiers caractères du titre

Lecteurs :
Le fichier doit mentionner :
- le numéro de la carte du lecteur
- les noms et adresse du lecteur
- la catégorie (éventuellement codée sur 3 caractères)
- prévoir une zone d'observations d'au moins une ligne
**- des codes pour les retards : nombre de jours de retards sur les 3 derniers mois

On doit pouvoir rechercher un lecteur par son nom ou son numéro de carte. Le résultat de la recherche doit être l'affichage des données suivantes concernant le lecteur :
- numéro de lecteur, nom et adresse
- nombre de livres empruntés et liste de ces livres comprenant la cote, le numéro d'inventaire, le début du titre, la date d'emprunt et la date de retour prévue.
- zone d'observations
**- livres réservés par le lecteur (numéro d'enregistrement, cote et titre sur 30 caractéres)
Même remarque que pour les documents lorsque la recherche donne plusieurs lecteurs (cas des homonymes).

Traitements :
- enregistrement d'un prêt
-retour d'un document prêté (avec signalement éventuel de la réservation)
- consultation de la situation d'un lecteur
-consultation de la situation d'un document
**- réservation d'un ouvrage pour un lecteur
**-édition de lettres de réclamations pour les ouvrages en retard de plus d'une semaine
**-statistiques :
Total du nombre de livres prêtés chaque mois
Total général en fin d'année
Nombre de livres prêtés par catégorie de lecteurs :
- par mois
- pour l'année
Nombre de livres prêtés par cote à x chiffres :
- par mois
- total annuel
Tableau du nombre de livres prêtés par grands secteurs et par catégories de lecteurs.
- chaque mois
- pour l'année

Sauvegardes :

Le système doit prévoir un journal réinitialisable.
Les sauvegardes doivent être fiables. Une procédure de récupération doit être prévue dès la version simplifiée.
Si la sauvegarde se fait sur disquettes, il faut prévoir l'extension des fichiers.

Editions :
- lettres de réclamation aux lecteurs pour plus d'une semaine de retard
- journal
**- liste des lecteurs avec paramétrage des critères suivants :
a) liste des noms ou liste des noms et des adresses ou liste des noms et des catégories
b) liste classée par numéro d'enregistrement ou par nom de lecteur
c) liste complète ou sélection d'une catégorie ou de plusieurs catégories de lecteurs
**- édition de la liste des lecteurs en retard avec mention des documents concernés et de leur date de retour prévue
**- édition de la liste des documents en retard avec le nom de l'emprunteur et la date de retour prévue
**- édition de la liste des ouvrages en prêt (pour me récolement) classée par cote

Le logiciel doit également prévoir un calendrier d'ouverture de la Bibliothèque pour la gestion des dates de retour.

PROGRAMME "BIBENS"

Les pages suivantes comprennent :

- La liste des fichiers-programmes
- La liste des fichiers bases de données
- La copie des 2 principaux menus
- Les listings des programmes dans l'ordre des choix des menus.

```
 Le volume dans l'unité C s'appelle DOS400
 Le numéro de série du volume est 0E4A-0ACA
 Répertoire de C:\DBASE

AM_SAUVE PRG        141 11/09/90    11:08
AM_MESSG PRG        897 11/09/90    11:30
AM_ANNRE PRG       1545 11/09/90    11:29
AM_RENOU PRG       2607 15/09/90    20:59
AM_RESER PRG       1893 14/09/90    22:03
BIBENS   PRG       2904 15/09/90    16:11
AM_LECSU PRG       3381 16/09/90    11:10
AM_SUPPR PRG        387 29/08/90    23:21
AM_PARAM PRG       2227 08/09/90    20:54
AM_ENRLE PRG       3079 15/09/90    13:01
AM_DOCSU PRG       1098 16/09/90    14:43
AM_ENRDO PRG       1975 15/09/90    13:02
AM_MODDO PRG       2489 08/09/90    21:25
AM_SITUL PRG       2629 15/09/90    19:15
AM_MODLE PRG       2318 08/09/90    21:12
AM_SITUD PRG       4269 06/09/90    17:14
AM_EDIT  PRG        748 15/09/90    10:20
AM_RETOU PRG       3149 16/09/90    12:00
AM_EDIPR PRG        916 08/09/90    21:31
AM_EDIRE PRG       1338 08/09/90    21:31
AM_EMPRU PRG       6550 15/09/90    15:32
AM_TRANS PRG        857 11/09/90    11:32
AM_STAT5 PRG       1322 15/09/90    14:30
AM_STAT  PRG        600 14/09/90    23:55
AM_AFFLE PRG        379 11/09/90    11:15
AM_AFFDO PRG        354 11/09/90    11:21
AM_MAJ   PRG        551 15/09/90    14:58
AM_RELAN PRG       3451 16/09/90    12:46
AM_STAT1 PRG       1020 15/09/90    15:06
AM_STAT2 PRG       2488 15/09/90    14:47
AM_STAT4 PRG       1374 15/09/90    21:29
AM_STAT3 PRG       1171 15/09/90    14:35
AM_FINRE PRG        241 16/09/90    11:14
AM_PENAL PRG        368 15/09/90    18:07
AM_EDILE PRG        250 15/09/90    10:39
       35 Fichier(s)   92051456 octets libres
```

Le volume dans l´unité C s´appelle DOS400
Le numéro de série du volume est OE4A-OACA
Répertoire de C:\DBASE\BIBENS

.		<REP>	03/09/90	7:43
..		<REP>	03/09/90	7:43
PRET	DBF	283	16/09/90	12:47
RESERVAT	DBF	130	04/09/90	14:59
LECTEUR	DBF	1448	16/09/90	12:47
SECCAT	DBF	1525	16/09/90	11:59
PRET	MDX	4097	16/09/90	11:58
DOCUMENT	MDX	8192	15/09/90	18:14
LECTEUR	MDX	6145	16/09/90	11:38
COMPTEUR	MDX	4097	15/09/90	18:29
DOCUMENT	DBF	871	16/09/90	11:59
COMPTEUR	DBF	502	16/09/90	11:59
PARAM	DBF	357	16/09/90	12:27
CATEG	DBF	734	16/09/90	11:38

14 Fichier(s) 92051456 octets libres

* MENU GENERAL *

1 TRANSACTIONS	8 RELANCES
2 ENREGISTREMENT LECTEUR	9 EDITIONS
3 SITUATION LECTEUR	10 STATISTIQUES
4 MODIFICATION LECTEUR	11 SAUVEGARDES
5 ENREGISTREMENT DOCUMENT	12 SUPPRESSIONS
6 SITUATION DOCUMENT	13 PARAMETRES
7 MODIFICATION DOCUMENT	14 FIN DE JOURNEE

Entrez le numéro choisi >

Durée de prêt 1 : 14   Durée de prêt 2 : 7   Nb maxi de prêts : 3
Prochaine fermeture : 23/12/90 - 04/01/91

* TRANSACTIONS *

1 EMPRUNT

2 RENOUVELLEMENT

3 RETOUR

4 RESERVATION

5 ANNULATION DE RESERVATION

6 MESSAGE

7 RETOUR MENU GENERAL

Entrez le numéro choisi >

Durée de prêt 1 : 14   Durée de prêt 2 : 7   Nb maxi de prêts : 3
Prochaine fermeture : 23/12/90 - 04/01/91

```
*BIBENS.PRG

PRIVATE CHOIX
PUBLIC DUR1,DUR2,PRMAX,FERM,OUV,INTREL,DURRES,DAT

CLEAR ALL

SET AUTOSAVE ON
SET BELL OFF
SET CONFIRM ON
SET ESCAPE OFF
SET PATH TO \DBASE\BIBENS
SET SCOREBOARD OFF
SET STATUS OFF
SET TALK OFF

DEFINE WINDOW cadre1 FROM 2,1 TO 19,39
DEFINE WINDOW cadre2 FROM 2,39 TO 19,78
DEFINE WINDOW sitlec FROM 6,21 TO 15,54
DEFINE WINDOW sitdoc FROM 6,21 TO 17,56
DEFINE WINDOW statis FROM 4,18 TO 18,60
DEFINE WINDOW supple FROM 9,10 TO 17,68
DEFINE WINDOW menu   FROM 3,1 TO 23,78
DEFINE WINDOW affstat FROM 3,2 TO 24,78 NONE

USE PARAM
DUR1  =DUREE1
DUR2  =DUREE2
PRMAX =MAXPRET
FERM  =DATEFERM
OUV   =DATEOUV
INTREL=INTERREL
DURRES=DUREERES
DAT   =MAJBASES
CLOSE ALL

CHOIX=SPACE(2)
DO WHILE CHOIX <> '14'
  DEACTIVATE WINDOW all
  CLEAR

  IF DAT<>DATE()
    IF YEAR(DAT)<>YEAR(DATE())
      DO AM_MAJ
    ENDIF
    USE PARAM
    DAT=DATE()
    REPLACE MAJBASES WITH DAT
    CLOSE ALL
    DO AM_FINRESER
    DO AM_PENAL
  ENDIF

  SET COLOR TO W
  ACTIVATE WINDOW menu
  @1,28 SAY '* MENU GENERAL *'
  SET COLOR TO BG
  @3,10 SAY  '1  TRANSACTIONS                8  RELANCES'
  @5,10 SAY  '2  ENREGISTREMENT LECTEUR      9  EDITIONS'
  @7,10 SAY  '3  SITUATION LECTEUR           10 STATISTIQUES'
  @9,10 SAY  '4  MODIFICATION LECTEUR        11 SAUVEGARDES'
  @11,10 SAY '5  ENREGISTREMENT DOCUMENT     12 SUPPRESSIONS'
  @13,10 SAY '6  SITUATION DOCUMENT          13 PARAMETRES'
  @15,10 SAY '7  MODIFICATION DOCUMENT       14 FIN DE JOURNEE'
  ACTIVATE SCREEN
  @24,1
```

Déclarer les variables

Paramétrer les commandes SET

Définir les fenêtres qui vont servir ultérieurement

Lire le fichier PARAM

DEBUT DE BOUCLE

Vérifier s'il est nécessaire de remettre à jour les compteurs statistiques et le cas échéant lancer AM_MAJ

Mettre à jour la date de mise en route

Effacer les réservations périmées

Mettre à jour les pénalisations

Afficher le menu général.

```
DO PARAMETRES

@19,21 SAY 'Entrez le numéro choisi >  ' GET CHOIX PICTURE '##'
READ
DEACTIVATE WINDOW menu

DO CASE
  CASE VAL(CHOIX) = 1
    DO AM_TRANS
  CASE CHOIX = '2'
    DO AM_ENRLEC
  CASE CHOIX = '3'
    DO AM_SITULEC
  CASE CHOIX = '4'
    DO AM_MODLEC
  CASE CHOIX = '5'
    VCOD=SPACE(7)
    DO AM_ENRDOC
  CASE CHOIX = '6'
    DO AM_SITUDOC
  CASE CHOIX = '7'
    DO AM_MODDOC
  CASE CHOIX = '8'
    DO AM_RELAN
  CASE CHOIX = '9'
    DO AM_EDIT
  CASE CHOIX = '10'
    DO AM_STAT
  CASE CHOIX = '11'
    DO AM_SAUVE
  CASE CHOIX = '12'
    DO AM_SUPPR
  CASE CHOIX = '13'
    DO AM_PARAMAJ
ENDCASE

ENDDO
CLEAR


                              &&PROCEDURES


Procedure FIGURE
ACTIVATE WINDOW cadre1
ACTIVATE WINDOW cadre2
ACTIVATE SCREEN
@3,5  SAY '[Lecteur]'
@3,65 SAY '[Document]'
RETURN


Procedure BEEP
SET BELL ON
?CHR(7)
SET BELL OFF
RETURN


Procedure PARAMETRES
?'    Durée de prêt 1 : ',LTRIM(STR(DUR1))
??'     Durée de prêt 2 : ',LTRIM(STR(DUR2)),'      Nb maxi de prêts : '
??LTRIM(STR(PRMAX))
?'                    Prochaine fermeture : ',FERM,'-',OUV
RETURN
```

Appeler la procédure d'affichage des paramètres

Lire le choix de l'utilisateur

Exécuter le programme correspondant au choix.

FIN DE BOUCLE

Afficher un cadre

Emettre un signal sonore

Afficher certains paramètres sous le menu.

```
Procedure PAUSE
DO WHILE INKEY()=0
ENDDO
RETURN
```

Attendre tant que l'utilisateur n'entre rien.

AM_TRANS.PRG

```
*AM_TRANS.PRG

PRIVATE CHOIX

CHOIX=SPACE(1)
DO WHILE CHOIX <> '7'                                         DEBUT DE BOUCLE
  ACTIVATE WINDOW menu
  SET COLOR TO W
  @1,28 SAY '* TRANSACTIONS *'                                Afficher le menu
  SET COLOR TO BG
  @3,23 SAY  '1  EMPRUNT'
  @5,23 SAY  '2  RENOUVELLEMENT'
  @7,23 SAY  '3  RETOUR'
  @9,23 SAY  '4  RESERVATION'
  @11,23 SAY '5  ANNULATION DE RESERVATION'
  @13,23 SAY '6  MESSAGE'
  @15,23 SAY '7  RETOUR MENU GENERAL'
  ACTIVATE SCREEN
  @24,1
  DO PARAMETRES

  @19,25 SAY 'Entrez le numéro choisi >  ' GET CHOIX PICTURE '#'     Lire le choix
  READ
  DEACTIVATE WINDOW trans
  DO CASE                                                     Exécuter le sous-programme
    CASE CHOIX = '1'                                          correspondant au choix.
      DO AM_EMPRUNT
    CASE CHOIX = '2'
      DO AM_RENOU
    CASE CHOIX = '3'
      DO AM_RETOUR
    CASE CHOIX = '4'
      DO AM_RESER
    CASE CHOIX = '5'
      DO AM_ANNRES
    CASE CHOIX = '6'
      DO AM_MESSG
  ENDCASE                                                     FIN DE BOUCLE
ENDDO

CLOSE ALL
CLEAR
RETURN
```

```
*AM_EMPRUNT.PRG

PRIVATE CARTE,REP,RESERVE,VRET1,AN,MOIS,VCAR,VNOM,VPRE,VCAT,VPEN,VNBP,VNBR,VMES,VVAL,VCOD,VCOT,VTIT,VINV,VTYP,VRES,VPR,VRET

CLEAR
DO FIGURE
SET COLOR TO W
@1,34 SAY '* Emprunt *'

SET COLOR TO BG
@22,1
DO PARAMETRES
CARTE=SPACE(3)
@5,5 SAY 'Code lecteur : ' GET CARTE PICTURE '###'
READ
VCAR=VAL(CARTE)
IF VCAR=0
  CLOSE ALL
  CLEAR
  RETURN
ENDIF

USE LECTEUR ORDER numcar
SEEK VCAR

IF EOF()
  CLOSE ALL
  @9,5 SAY 'Lecteur non enregistré...>'
  DO BEEP
  DO PAUSE
  CLEAR
  RETURN
ENDIF

VNOM=NOM
VPRE=PRENOM
VCAT=CATEG
VPEN=PENAL
VNBP=NBPRET
VNBR=NBRESER
VMES=MESSG
VVAL=VALID
CLOSE ALL

@6,5 SAY 'Nom    : '
??TRIM(VNOM)
@7,5 SAY 'Prénom : '
??TRIM(VPRE)
@8,5 SAY 'Catég. : '
??VCAT
IF VMES <> SPACE(160)
  SET COLOR TO W
  @20,1 SAY 'MESSAGE : '
  SET COLOR TO BG
  ??VMES
ENDIF

IF VVAL<DATE()
  @10,5 SAY 'Inscription périmée, prêt impossible...>'
  DO BEEP
  DO PAUSE
  CLOSE ALL
  CLEAR
  RETURN
ENDIF
```

Afficher paramètres en bas d'écran.
Lire le code lecteur
Si aucun code n'est lu, quitter

Vérifier si le code lecteur existe

Si oui :
Lire les coordonnées du lecteur

Afficher les coordonnées du lecteur

Vérifier s'il est encore inscrit

Vérifier si le lecteur est pénalisé

```
IF VPEN>DATE()
  @10,5 SAY 'Pénalisé jusqu"au : '
  ??VPEN
  @11,5 SAY 'Lecteur pénalisé, prêt impossible...>'
  DO BEEP
  DO PAUSE
  CLOSE ALL
  CLEAR
  RETURN
ENDIF
```

DEBUT DE BOUCLE

Afficher son nombre de prêts

Signaler si la limite est atteinte

Proposer une autorisation

```
REP='O'
DO WHILE UPPER(REP)<>'N'
  @12,5 SAY 'Prêts : '
  ??LTRIM(STR(VNBP))
  IF VNBP>=PRMAX
    @14,5 SAY 'Nombre maximum de prêts atteint'
    DO BEEP
    REP='N'
    @15,5 SAY 'Autorisez-vous le prêt (O/N) ? ' GET REP
    READ
    @15,5 CLEAR TO 15,38
    IF UPPER(REP)='N'
      CLOSE ALL
      CLEAR
      RETURN
    ENDIF
  ENDIF
```

Lire le code du document

Si aucun code n'est entré, quitter

```
  VCOD=SPACE(7)
  @5,43 SAY 'Code document : ' GET VCOD PICTURE '999999X' FUNCTION '!'
  READ
  IF VCOD=SPACE(7)
    CLOSE ALL
    CLEAR
    RETURN
  ENDIF
```

Vérifier si ce code existe

```
  USE DOCUMENT ORDER codbarr
  SEEK VCOD
  IF EOF()
    CLOSE ALL
    DO BEEP
    @9,43 SAY 'Document non enregistré...>'
    DO PAUSE
```

Si oui :

Lire les coordonnées du document

```
  ELSE
    VTIT=TITRE
    VCOT=COTE
    VINV=NUMINV
    VTYP=TYPDOC
    VRES=RESER
    VPR =PRET
    CLOSE ALL
```

Vérifier s'il est en prêt

```
    IF VPR<>0
      @9,43 SAY 'Document en prêt...>'
      DO BEEP
      DO PAUSE
    ELSE
```

Vérifier s'il est réservé

Si oui :

Voir s'il s'agit de l'emprunteur

```
      RESERVE=0
      IF VRES<>0
        USE RESERVAT
        LOCATE FOR VCOD=CODBARR
        IF NUMCAR<>VCAR
```

```
    CLOSE ALL
    RESERVE=1
    @9,43 SAY 'Document réservé...>'
    DO BEEP
    DO PAUSE
  ELSE
    DELETE
    PACK
    VRES=VRES-1
    VNBR=VNBR-1
    CLOSE ALL
  ENDIF
ENDIF

IF RESERVE=0
  @6,43 SAY 'Cote  : '
  ??VCOT
  @7,43 SAY 'Titre : '
  ??SUBSTR(VTIT,1,25)
  @8,43 SAY 'N°inv.: '
  ??VINV

  IF VTYP=2
    VRET=DATE()+DUR2
  ELSE
    VRET=DATE()+DUR1
  ENDIF
  IF VRET>=FERM .AND. VRET<=OUV
    VRET=OUV+1
  ENDIF
  VRET1=VRET

  @11,43 SAY 'Date de retour : ' GET VRET FUNCTION 'D'
  @13,47 SAY 'Validez ou modifiez.'
  READ
  DO WHILE VRET<=DATE() .OR. VRET={  /  /  }
    VRET=VRET1
    DO BEEP
    @11,61 GET VRET FUNCTION 'D'
    READ
  ENDDO
  @13,47 CLEAR TO 13,76

  USE PRET
  APPEND BLANK
  REPLACE NUMCAR WITH VCAR
  REPLACE CODBARR WITH VCOD
  REPLACE DATESOR WITH DATE()
  REPLACE DATERET WITH VRET

  VNBP=VNBP+1
  USE LECTEUR ORDER numcar
  SEEK VCAR
  REPLACE NBRESER WITH VNBR
  REPLACE NBPRET WITH VNBP
  @12,13 SAY NBPRET

  IF VNBP=1
    USE PARAM
    REPLACE ACTIFS WITH ACTIFS+1
  ENDIF

  USE DOCUMENT ORDER codbarr
  SEEK VCOD
  REPLACE RESER WITH VRES
  REPLACE PRET WITH 1
```

S'il ne s'agit pas de l'emprunteur, signaler la réservation, attendre, quitter.

S'il s'agit de l'emprunteur, effacer sa réservation

Si non-réservé

afficher les coordonnées du document.

Calculer la date de retour

Afficher la date de retour
Proposer la modification éventuelle.
Vérifier si la date modifiée est correcte

Enregistrer le prêt

Mettre à jour les compteurs 'réservation' et 'prêt' de l'emprunteur

Mettre à jour le compteur du nombre de lecteurs actifs

Mettre à jour les compteurs 'réservation' et 'prêt' du document,

```
REPLACE AN1 WITH AN1+1

USE SECCAT
DO CASE
  CASE LIKE('P*',VCOT)
    GO 1
  CASE LIKE('A*',VCOT) .AND. (.NOT. LIKE('AB*',VCOT))
    GO 2
  CASE LIKE('B*',VCOT) .AND. (.NOT. LIKE('BA*',VCOT)) .AND. (.NOT. LIKE('BB*',VCOT)) .AND. (.NOT. LIKE('BC*',VCOT))
    GO 3
  CASE LIKE('C*',VCOT) .AND. (.NOT. LIKE('CA*',VCOT))
    GO 4
  CASE LIKE('AB*',VCOT)
    GO 5
  CASE LIKE('BA*',VCOT)
    GO 6
  CASE LIKE('BB*',VCOT)
    GO 7
  CASE LIKE('BC*',VCOT)
    GO 8
  CASE LIKE('CA*',VCOT)
    GO 9
  CASE LIKE('W*',VCOT)
    GO 10
  CASE LIKE('X*',VCOT)
    GO 11
  CASE LIKE('Y*',VCOT)
    GO 12
  CASE LIKE('Z*',VCOT)
    GO 13
  OTHERWISE
    @20,5 SAY 'Le secteur '+VCOT+'n est pas prévu...>'
    DO BEEP
    DO PAUSE
    CLOSE ALL
    CLEAR
    RETURN
ENDCASE
DO CASE
  CASE VCAT='A'
    REPLACE CATA WITH CATA+1
  CASE VCAT='B'
    REPLACE CATB WITH CATB+1
  CASE VCAT='C'
    REPLACE CATC WITH CATC+1
  CASE VCAT='D'
    REPLACE CATD WITH CATD+1
  CASE VCAT='E'
    REPLACE CATE WITH CATE+1
  CASE VCAT='F'
    REPLACE CATF WITH CATF+1
  CASE VCAT='G'
    REPLACE CATG WITH CATG+1
  CASE VCAT='H'
    REPLACE CATH WITH CATH+1
  CASE VCAT='I'
    REPLACE CATI WITH CATI+1
  CASE VCAT='J'
    REPLACE CATJ WITH CATJ+1
  CASE VCAT='K'
    REPLACE CATK WITH CATK+1
  CASE VCAT='L'
    REPLACE CATL WITH CATL+1
  CASE VCAT='M'
    REPLACE CATM WITH CATM+1
  CASE VCAT='N'
```

Mettre à jour le compteur 'des prêts de l'année' du document

Mettre à jour le compteur des prêts par secteur documentaire

Signaler si le secteur n'existe pas

Mettre à jour le compteur des prêts par catégories de lecteur

```
      REPLACE CATN WITH CATN+1
    CASE VCAT='O'
      REPLACE CATO WITH CATO+1
    CASE VCAT='P'
      REPLACE CATP WITH CATP+1
    CASE VCAT='Q'
      REPLACE CATQ WITH CATQ+1
    CASE VCAT='R'
      REPLACE CATR WITH CATR+1
    CASE VCAT='S'
      REPLACE CATS WITH CATS+1
    CASE VCAT='T'
      REPLACE CATT WITH CATT+1
    OTHERWISE
      @20,5 SAY 'La catégorie '+VCAT+'n est pas prévue...>'
      DO BEEP
      DO PAUSE
      CLOSE ALL
      CLEAR
      RETURN
  ENDCASE

  AN=YEAR(DATE())
  USE COMPTEUR ORDER année
  SEEK AN
  IF EOF()
    APPEND BLANK
    REPLACE ANNEE WITH AN
  ENDIF
  MOIS=MONTH(DATE())
  DO CASE
    CASE MOIS=1
      REPLACE JAN WITH JAN+1
    CASE MOIS=2
      REPLACE FEV WITH FEV+1
    CASE MOIS=3
      REPLACE MAR WITH MAR+1
    CASE MOIS=4
      REPLACE AVR WITH AVR+1
    CASE MOIS=5
      REPLACE MAI WITH MAI+1
    CASE MOIS=6
      REPLACE JUN WITH JUN+1
    CASE MOIS=7
      REPLACE JUL WITH JUL+1
    CASE MOIS=8
      REPLACE AOU WITH AOU+1
    CASE MOIS=9
      REPLACE SEP WITH SEP+1
    CASE MOIS=10
      REPLACE OCT WITH OCT+1
    CASE MOIS=11
      REPLACE NOV WITH NOV+1
    CASE MOIS=12
      REPLACE DEC WITH DEC+1
  ENDCASE

  CLOSE ALL

  SET CONSOLE OFF
  SET PRINT ON
  ?DATE(),'     Emprunt.'
  ?'    Code document : ',VCOD
  ?'    Code lecteur  : ',VCAR
  ?'    Date retour   : ',VRET
  ?'    *'
```

Signaler si la catégorie n'existe pas.

Mettre à jour le compteur de prêts mensuels.
Si : l'année a changé, ajouter un nouvel enregistrement au fichier COMPTEUR

Imprimer les informations essentielles.

```
        ?' '
        SET PRINT OFF
        SET CONSOLE ON
      ENDIF
    ENDIF
  ENDIF

  @9,43 CLEAR TO 9,76
  REP='O'
  @17,43 SAY 'Autre emprunt (O/N) ? ' GET REP
  READ

  @11,5 CLEAR TO 11,15
  @5,43 CLEAR TO 5,76
  @6,43 CLEAR TO 6,76
  @7,43 CLEAR TO 7,76
  @8,43 CLEAR TO 8,76
  @11,43 CLEAR TO 11,76
  @17,43 CLEAR TO 17,76
ENDDO

CLEAR
RETURN
```

Proposer l'enchaînement sur un autre emprunt.

Effacer le contenu du cadre de droite

FIN DE BOUCLE

```
*AM_RENOU.PRG

PRIVATE REP

CLEAR
DO FIGURE
SET COLOR TO W
@1,28 SAY '* Renouvellement de prêt *'
SET COLOR TO BG
@22,1
DO PARAMETRES

VINV=SPACE(8)
@5,43 SAY 'N° inventaire : ' GET VINV PICTURE '#####X##'
READ
IF VINV=SPACE(8)
  CLOSE ALL
  CLEAR
  RETURN
ENDIF

USE DOCUMENT ORDER numinv
SEEK VINV
IF EOF()
  @9,45 SAY 'Document non enregistré...>'
  DO BEEP
  DO PAUSE
  CLOSE ALL
  CLEAR
  RETURN
ENDIF
TIT=TITRE
VCOD=CODBARR
VTYP=TYPDOC
VPR=PRET
CLOSE ALL
IF VPR=0
  @9,45 SAY 'Document non prêté...>'
  DO BEEP
  CLOSE ALL
  CLEAR
  RETURN
ENDIF
@6,43 SAY 'Type  : '
??VTYP
@7,43 SAY 'Titre : '
??SUBSTR(TIT,1,27)

REP='O'
@9,45 SAY 'Continuer (O/N) ? ' GET REP
READ
IF UPPER(REP)='N'
  CLOSE ALL
  CLEAR
  RETURN
ENDIF
@9,45 CLEAR TO 9,76

USE RESERVAT
LOCATE FOR CODBARR=VCOD
IF CODBARR=VCOD
  @9,45 SAY 'Ce document est réservé...>'
  DO BEEP
  DO PAUSE
  CLOSE ALL
  CLEAR
```

Lire le n° inventaire

Si aucun n° n'est entré, quitter

Vérifier si le document existe

Si oui :

Lire ses coordonnées

Vérifier s'il est bien emprunté

Afficher ses coordonnées

Attendre confirmation par l'utilisateur

Vérifier si le document est réservé

```
  RETURN
ENDIF
@9,45 CLEAR TO 9,76

USE PRET
LOCATE FOR CODBARR=VCOD
VCAR=NUMCAR
VRET=DATERET
VREN=RENOU
CLOSE ALL
@8,43 SAY 'Retour prévu : '
??VRET
IF VRET<DATE()
  @10,41 SAY 'Retard, renouvellement impossible...>'
  DO BEEP
  DO PAUSE
  CLOSE ALL
  CLEAR
  RETURN
ENDIF
USE LECTEUR ORDER numcar
SEEK VCAR
VNOM=NOM
VPRE=PRENOM
VCAT=CATEG
CLOSE ALL
@5,5 SAY 'Code lecteur : '
??VCAR
@6,5 SAY 'Nom    : '
??VNOM
@7,5 SAY 'Prénom : '
??VPRE
@8,5 SAY 'Catég. : '
??VCAT

IF VTYP=1 .AND. (VREN=1 .OR. VREN=2)
  @17,5 SAY 'Prêt déjà renouvelé'
  DO BEEP
  REP='N'
  @18,3 SAY 'Accordez-vous le prolongement (O/N) ? ' GET REP
  READ
  @18,3 CLEAR TO 18,38
  IF UPPER(REP)='N'
    CLOSE ALL
    CLEAR
    RETURN
  ENDIF
ELSE
  IF VREN=3
    @17,5 SAY 'Prêt déjà renouvelé 3 fois...>'
    DO BEEP
    DO PAUSE
    CLOSE ALL
    CLEAR
    RETURN
  ENDIF
ENDIF
IF VTYP=1
  VRET=VRET+DUR1
ELSE
  VRET=VRET+DUR2
ENDIF
IF VRET<=FERM .AND. VRET<=OUV
  VRET=OUV+1
ENDIF
VRET1=VRET
```

Lire l'enregistrement correspondant au prêt
Vérifier si le lecteur est en retard

Signaler le retard éventuel, attendre, quitter.

Lire l'enregistrement du lecteur

Afficher ses coordonnées

Dans le cas où le document est de type 1 et a été renouvelé une ou deux fois, le signaler
Proposer une décision de prolonger encore le prêt

Dans le cas où le document est de type 2, signaler si le prêt a déjà été prolongé 3 fois, attendre, quitter

Si le renouvellement est possible, Calculer la nouvelle date de retour

```
@12,43 SAY 'Renouvelé jusqu au : ' GET VRET FUNCTION 'D'
@13,47 SAY 'Validez ou modifiez.'
READ
DO WHILE VRET<=DATE() .OR. VRET={  /  /  }
  VRET=VRET1
  DO BEEP
  @11,61 GET VRET FUNCTION 'D'
  READ
ENDDO
@13,47 CLEAR TO 13,76
VREN=VREN+1
USE PRET
LOCATE FOR NUMCAR=VCAR
REPLACE DATERET WITH VRET
REPLACE RENOU WITH VREN
CLOSE ALL

SET CONSOLE OFF
SET PRINT ON
?DATE(),'    Renouvellement de prêt ',VREN
?'      Code lecteur     : ',VCAR
?'      N°inventaire     : ',VINV
?'      Renouv. jusqu au : ',VRET
?'      *'
?' '
SET PRINT OFF
SET CONSOLE ON

CLEAR
RETURN
```

Afficher cette date et en proposer la modification éventuelle
Vérifier si la modification est une date correcte

Modifier l'enregistrement de ce prêt.

Imprimer les informations essentielles

```
*AM_RETOUR.PRG

PRIVATE REP

CLEAR
DO FIGURE
SET COLOR TO W
@1,35 SAY '* Retour *'

SET COLOR TO BG
@22,1
DO PARAMETRES
REP='O'
DO WHILE UPPER(REP)<>'N'
  VCOD=SPACE(7)
  @5,43 SAY 'Code document : ' GET VCOD PICTURE '999999X' FUNCTION '!'
  READ
  IF VCOD=SPACE(7)
    CLOSE ALL
    CLEAR
    RETURN
  ENDIF

  USE DOCUMENT ORDER codbarr
  SEEK VCOD
  IF EOF()
    CLOSE ALL
    @9,43 SAY 'Document non enregistré...>'
    DO BEEP
    DO PAUSE
  ELSE
    VCOT=COTE
    VTIT=TITRE
    VINV=NUMINV
    VRES=RESER
    VPR =PRET
    CLOSE ALL
    @6,43 SAY 'Cote  : '
    ??VCOT
    @7,43 SAY 'Titre : '
    ??SUBSTR(VTIT,1,25)
    @8,43 SAY 'N°inv.: '
    ??VINV

    USE PRET
    LOCATE FOR CODBARR=VCOD
    IF EOF()
      CLOSE ALL
      @9,43 SAY 'Document non prêté...>'
      DO BEEP
      DO PAUSE
    ELSE
      VCAR=NUMCAR
      VRET=DATERET
      DELETE
      PACK

      USE DOCUMENT ORDER codbarr
      SEEK VCOD
      REPLACE PRET WITH 0

      CLOSE ALL

      IF VRET<DATE()
        @10,43 SAY 'Retour prévu : '
        ??VRET
```

DEBUT DE BOUCLE

Lire le code du document

Si aucun code n'est entré, quitter

Vérifier si ce document existe

Si oui :
Lire ses coordonnées

Afficher ses coordonnées

Vérifier si le document a bien été emprunté

Si oui :
Lire le code lecteur
Lire la date de retour
Effacer l'enregistrement du prêt.
Mettre à jour le compteur 'prêt' du document

En cas de retard, afficher la date de retour prévue (dépassée)

```
ENDIF
```

Si le compteur 'réservation' du document est égal à 0, affecter des valeurs nulles ou vides aux variables

```
IF VRES=0
  XCAR=0
  XLIM={ / / }
  XNOM=SPACE(25)
  XPRE=SPACE(25)
  XRES=SPACE(40)
  XRUE=SPACE(40)
  XCOP=SPACE(5)
  XLIE=SPACE(15)
```

Sinon, lire dans le fichier RESERVAT
le code lecteur
le code document
calculer la date limite de réservation

```
ELSE
  USE RESERVAT
  LOCATE FOR CODBARR=VCOD
  XCAR=NUMCAR
  XLIM=DATE()+DURRES
  REPLACE DATELIM WITH XLIM
  CLOSE ALL
```

lire les coordonnées du lecteur dans le fichier LECTEUR (=lecteur qui a réservé le document retourné)

```
  USE LECTEUR ORDER numcar
  SEEK XCAR
  XNOM=NOM
  XPRE=PRENOM
  XRES=RESID
  XRUE=RUE
  XCOP=CODPOST
  XLIE=LIEU
```

Afficher nom et initiale du prénom du lecteur qui a réservé
Afficher la date limite de réservation

```
  @13,47 SAY ´Réserver pour : ´
  ??TRIM(SUBSTR(XNOM,1,14)),SUBSTR(XPRE,1,1),´.´
  @14,47 SAY ´Jusqu au        : ´
  ??XLIM
ENDIF
```

Dans le fichier LECTEUR, lire les coordonnées du lecteur qui rend le document
mettre à jour son compteur 'prêt'

```
USE LECTEUR ORDER numcar
SEEK VCAR
VNOM=NOM
VPRE=PRENOM
VCAT=CATEG
VMES=MESSG
VPEN=PENAL
REPLACE NBPRET WITH NBPRET-1
VNBP=NBPRET
```

Mettre à jour le compteur des lecteurs actifs

```
IF VNBP=0
  USE PARAM
  REPLACE ACTIFS WITH ACTIFS-1
ENDIF
CLOSE ALL
```

Afficher les coordonnées du lecteur qui rend le document

```
@5,5 SAY ´Code lecteur : ´
??VCAR
@6,5 SAY ´Nom     : ´
??TRIM(VNOM)
@7,5 SAY ´Prénom : ´
??TRIM(VPRE)
@8,5 SAY ´Catég. : ´
??VCAT
IF VMES<>SPACE(160)
  SET COLOR TO W
  @20,1 SAY ´MESSAGE : ´
  ??VMES
  SET COLOR TO BG
ENDIF
```

Afficher le nombre actualisé de ses prêts

```
@10,5 SAY ´Prêts restant : ´
??VNBP
```

Afficher une pénalisation éventuelle

```
IF VPEN>DATE()
  @11,5 SAY ´Pénalisé jusqu au : ´
```

```
      ??VPEN
     ENDIF

     SET CONSOLE OFF
     SET PRINT ON
     ?DATE(),'     Retour'
     ?'    Code document : ',VCOD
     ?'    Code lecteur  : ',VCAR
     ?'    Pénalisation  : ',VPEN
     IF XCAR<>0
       ?'    Réserver pour : ',XCAR
       ?'    Nom           : ',XNOM
       ?'    Prénom        : ',XPRE
       ?'    Adresse       : ',XRES
       ?'                    ',XRUE
       ?'                    ',XCOP,XLIE
       ?'    Jusqu au      : ',XLIM
     ENDIF
     ?'    *'
     ?' '
     SET PRINT OFF
     SET CONSOLE ON
   ENDIF
 ENDIF
 @9,43 CLEAR TO 9,76
 REP='O'
 @17,43 SAY 'Autre retour (O/N) ? ' GET REP
 READ
 @5,43 CLEAR TO 5,76
 @6,43 CLEAR TO 6,76
 @7,43 CLEAR TO 7,76
 @8,43 CLEAR TO 8,76
 @10,43 CLEAR TO 10,76
ENDDO

CLEAR
RETURN
```

Imprimer les informations essentielles

Si nécessaire, les coordonnées du lecteur qui a réservé le document retourné.

Proposer l'enchaînement sur un autre retour

Effacer les données du cadre de droite

FIN DE BOUCLE

```
*AM_RESER.PRG

 PRIVATE REP

 CLEAR
 DO FIGURE
 SET COLOR TO W
 @1,33 SAY '* Réservation *'

 SET COLOR TO BG
 @22,1
 DO PARAMETRES
 VINV=SPACE(8)
 @5,43 SAY 'N° inventaire : ' GET VINV PICTURE '#####X##'
 READ
 IF VINV=SPACE(8)
   CLOSE ALL
   CLEAR
   RETURN
 ENDIF

 USE DOCUMENT ORDER numinv
 SEEK VINV
 IF EOF()
   @7,43 SAY 'Document non enregistré...>'
   DO BEEP
   DO PAUSE
   CLOSE ALL
   CLEAR
   RETURN
 ENDIF
 TIT=TITRE
 VCOD=CODBARR
 @6,43 SAY SUBSTR(TIT,1,30)
 IF RESER=3
   @7,43 SAY 'Limite de 3 réservations atteinte...>'
   DO BEEP
   DO PAUSE
   CLOSE ALL
   CLEAR
   RETURN
 ELSE
   VRES=RESER+1
   CLOSE ALL
 ENDIF
 REP='O'
 @8,45 SAY 'Continuer O/N ? ' GET REP
 READ
 IF UPPER(REP)='N'
   CLOSE ALL
   CLEAR
   RETURN
 ELSE
   @8,45 CLEAR TO 8,76


   CARTE=SPACE(3)
   @5,5 SAY 'Code lecteur : ' GET CARTE PICTURE '###'
   READ
   VCAR=VAL(CARTE)
   IF VCAR=0
     CLOSE ALL
     CLEAR
     RETURN
   ENDIF
```

Lire le n° inventaire du document
Si aucun n° n'est entré, quitter

Vérifier si ce document existe

Si oui, afficher son titre

Vérifier s'il a déjà fait l'objet de 3 réservations
si oui, attendre, quitter

si non,

demander confirmation à l'utilisateur

Lire le code du lecteur
Si aucun code n'est entré, quitter

```
  USE LECTEUR ORDER numcar
  SEEK VCAR
  IF EOF()
    @9,5 SAY 'Lecteur non enregistré...>'
    DO BEEP
    DO PAUSE
    CLOSE ALL
    CLEAR
    RETURN
  ENDIF

  USE RESERVAT
  LOCATE FOR NUMCAR=VCAR .AND. CODBARR=VCOD
  IF .NOT. EOF()
    @8,5 SAY 'Déjà réservé par ce lecteur...>'
    DO BEEP
    DO PAUSE
    CLOSE ALL
    CLEAR
    RETURN
  ENDIF

  APPEND BLANK
  REPLACE NUMCAR WITH VCAR
  REPLACE CODBARR WITH VCOD

  USE LECTEUR ORDER numcar
  SEEK VCAR
  REPLACE NBRESER WITH NBRESER+1
  @6,5 SAY 'Nom    : '
  ??NOM
  @7,5 SAY 'Prénom : '
  ??PRENOM
  @9,8 SAY 'Réservations : '
  ??NBRESER

  USE DOCUMENT ORDER codbarr
  SEEK VCOD
  REPLACE RESER WITH VRES
  CLOSE ALL

  SET CONSOLE OFF
  SET PRINT ON
  ?DATE(),'    Réservation.'
  ?'   N°inventaire     : ',VINV
  ?'   Code lecteur     : ',VCAR
  ?'   *'
  ?' '
  SET PRINT OFF
  SET CONSOLE ON
ENDIF

CLEAR
RETURN
```

Vérifier si le lecteur existe

Vérifier si le lecteur a déjà réservé ce document (réservation en cours)

Si non,
Enregistrer sa réservation

Mettre à jour le compteur 'reservation' du lecteur
Afficher ses coordonnées

Afficher le nombre actualisé de ses réservations

Mettre à jour le compteur 'reservation' du document

Afficher les informations utiles.

```
*AM_ANNRES.PRG

PRIVATE REP

CLEAR
DO FIGURE
SET COLOR TO W
@1,25 SAY '* Annulation de réservation *'

SET COLOR TO BG
@22,1
DO PARAMETRES
VINV=SPACE(8)
@5,43 SAY 'N° inventaire : ' GET VINV PICTURE '#####X##'
READ
IF VINV=SPACE(8)
  CLOSE ALL
  CLEAR
  RETURN
ENDIF

USE DOCUMENT ORDER numinv
SEEK VINV
IF EOF()
  CLOSE ALL
  @9,45 SAY 'Document non enregistré...>'
  DO BEEP
  DO PAUSE
  CLEAR
  RETURN
ENDIF
TIT=TITRE
@6,43 SAY SUBSTR(TIT,1,30)
REP='0'
@8,45 SAY 'Continuer (O/N) ? ' GET REP
READ
IF UPPER(REP)='N'
  CLOSE ALL
  CLEAR
  RETURN
ELSE
  @8,45 CLEAR TO 8,77
  VRES=RESER-1
  VCOD=CODBARR
  CLOSE ALL

  CARTE=SPACE(3)
  @5,5 SAY 'Code lecteur : ' GET CARTE PICTURE '###'
  READ
  VCAR=VAL(CARTE)
  IF VCAR=0
    CLOSE ALL
    CLEAR
    RETURN
  ENDIF

  USE RESERVAT
  LOCATE FOR NUMCAR=VCAR .AND. CODBARR=VCOD
  IF EOF()
    CLOSE ALL
    @8,5 SAY 'Réservation inexistante...>'
    DO BEEP
    DO PAUSE
    CLEAR
    RETURN
  ENDIF
```

Lire le n° inventaire du document

Si aucun no n'est entré, quitter

Vérifier si le n° correspond bien à un document enregistré

Affiche le titre correspondant

Demander confirmation

Lire le compteur 'réservation' du document et le mettre à jour (variable)

Lire le code du document

Lire le code du lecteur

Si aucun code n'est rentré, quitter

Vérifier l'existence de la réservation de ce document par ce lecteur

```
  DELETE
  PACK

  USE LECTEUR ORDER numcar
  SEEK VCAR
  REPLACE NBRESER WITH NBRESER-1
  @6,5 SAY ´Nom    : ´
  ??NOM
  @7,5 SAY ´Prénom : ´
  ??PRENOM
  @9,5 SAY ´Réservations : ´
  ??NBRESER

  USE DOCUMENT ORDER codbarr
  SEEK VCOD
  REPLACE RESER WITH VRES
  CLOSE ALL

  SET CONSOLE OFF
  SET PRINT ON
  ?DATE(),´    Annulation de réservation.´
  ?´   N°inventaire    : ´,VCOD
  ?´   Code lecteur    : ´,VCAR
  ?´   *´
  ?´ ´
  SET PRINT OFF
  SET CONSOLE ON

ENDIF

CLEAR
RETURN
```

Mettre à jour le compteur 'réservation' du lecteur

Afficher ses coordonnées

Afficher le nombre actualisé de ses réservations

Mettre à jour le compteur 'réservation' du document en le remplaçant par la variable.

Imprimer les informations essentielles

AM_MESSG . PRG

```
*AM_MESSG.PRG

PRIVATE CARTE,REP

CLEAR
DO FIGURE
SET COLOR TO W
@1,29 SAY '* Message à lecteur *'

SET COLOR TO BG
@22,1
DO PARAMETRES
CARTE=SPACE(3)
@5,5 SAY 'Code lecteur : ' GET CARTE PICTURE '###'      Lire le code lecteur
READ
VCAR=VAL(CARTE)

USE LECTEUR ORDER numcar                                 Vérifier si ce lecteur existe
SEEK VCAR
IF EOF()
  CLOSE ALL
  @6,5 SAY 'Lecteur non enregistré...>'
  DO BEEP
  DO PAUSE
  CLEAR
  RETURN
ENDIF                                                    Lire ses coordonnées
VCAR=NUMCAR
VNOM=NOM                                                 Afficher ses coordonnées
VPRE=PRENOM
@6,5 SAY 'Nom    : '
??NOM
@7,5 SAY 'Prénom : '
??PRENOM
@8,5 SAY 'Catég. : '
??CATEG
VMES=MESSG
@17,5 SAY 'Entrez ou modifiez le message (2 lignes maximum): '   Lire le message entré
@18,2 GET VMES
READ
REPLACE MESSG WITH VMES                                  Remplacer le champ 'message' par
CLOSE ALL                                                la nouvelle valeur

SET CONSOLE OFF
SET PRINT ON
?DATE(),'     Message à : '                              Imprimer les informations
?'   Code lecteur : ',VCAR                               essentielles
?'   Nom          : ',TRIM(VNOM)
?'   Prénom       : ',TRIM(VPRE)
?VMES
?'   *'
?' '
SET PRINT OFF
SET CONSOLE ON

CLOSE ALL
CLEAR
RETURN
```

```
*AM_ENRLEC.PRG

PRIVATE REP,N

CLEAR
SET COLOR TO W
@2,28 SAY '* ENREGISTREMENT LECTEUR *'
SET COLOR TO BG

USE LECTEUR  ORDER numcar
N=1
DO WHILE .NOT. EOF() .AND. N<1000
  N=N+1
  LOCATE FOR NUMCAR=N   SEEK N
ENDDO
CLOSE ALL
```

Rechercher dans le fichier LECTEUR un numéro de code inexistant

```
IF N=1000
  @6,5 SAY 'Capacité en lecteurs atteinte (999)...>'
  DO BEEP
  DO PAUSE
  CLEAR
  RETURN
ENDIF
```

<u>Si</u> l'incrémentation de N arrive à 1000, impossible d'enregistrer un nouveau lecteur,
le signaler par un message, attendre, quitter

```
VCAR=N
VCAT=SPACE(1)
VNOM=SPACE(20)
VPRE=SPACE(20)
VRES=SPACE(40)
VRUE=SPACE(40)
VCOP=SPACE(5)
VLIE=SPACE(20)
VNBP=0
VNBR=0
VMES=SPACE(160)
VVAL=DATE()+364
```

<u>Sinon</u>, attribuer la valeur de N à une variable
initialiser toutes les autres variables

```
@5,5 SAY 'Code          : ' + STR(VCAR)
@6,5 SAY 'Nom           : ' GET VNOM FUNCTION '!'
@7,5 SAY 'Prénom        : ' GET VPRE FUNCTION '!'
READ
```

Afficher le code
Lire le nom, le prénom

```
REP='N'
USE LECTEUR
SCAN FOR NOM=VNOM .AND. PRENOM=VPRE
  @13,26 SAY 'Nom        : '+ NOM
  @14,26 SAY 'Prénom     : '+ PRENOM
  @15,26 SAY 'Adresse    : '+ RESID
  @16,26 SAY '             '+ RUE
  @17,26 SAY '             '+ CODPOST
  @17,45 SAY LIEU
  @19,26 SAY 'Est-ce le même lecteur (O/N) ? ' GET REP
  READ
  IF UPPER(REP)='O'
    WAIT 'Appuyez sur une touche pour revenir au menu...>'
    CLOSE ALL
    CLEAR
    RETURN
  ENDIF
  @13,26 CLEAR
ENDSCAN
CLOSE ALL
```

Rechercher s'il existe des homonymes
afficher, le cas échéant leurs coordonnées

Demander une vérification à l'utilisateur
Attendre, retourner au menu (cas où il s'agit du même lecteur)

```
@8,5  SAY 'Résidence   : ' GET VRES
@9,5  SAY 'Rue         : ' GET VRUE
```

Lire les coordonnées suivantes

AM_ENRLEC.PRG 2

```
@10,5 SAY 'Code postal : ' GET VCOP PICTURE '#####'
@11,5 SAY 'Localité    : ' GET VLIE FUNCTION '!'
@12,5 SAY 'Catégorie   : ' GET VCAT FUNCTION '!'
@13,5 SAY 'Observation : ' GET VMES
@16,5 SAY 'Validité    : ' GET VVAL FUNCTION 'D'
READ

REP='N'
@18,5 SAY 'Validation (O/N) ? ' GET REP
READ
IF UPPER(REP)='O'
  USE LECTEUR
  APPEND BLANK
  REPLACE NUMCAR WITH VCAR
  REPLACE CATEG WITH VCAT
  REPLACE NOM WITH VNOM
  REPLACE PRENOM WITH VPRE
  REPLACE RESID WITH VRES
  REPLACE RUE WITH VRUE
  REPLACE CODPOST WITH VCOP
  REPLACE LIEU WITH VLIE
  REPLACE MESSG WITH VMES
  REPLACE VALID WITH VVAL

  USE CATEG
  DO CASE
    CASE VCAT='A'
      REPLACE CATA WITH CATA+1
    CASE VCAT='B'
      REPLACE CATB WITH CATB+1
    CASE VCAT='C'
      REPLACE CATC WITH CATC+1
    CASE VCAT='D'
      REPLACE CATD WITH CATD+1
    CASE VCAT='E'
      REPLACE CATE WITH CATE+1
    CASE VCAT='F'
      REPLACE CATF WITH CATF+1
    CASE VCAT='G'
      REPLACE CATG WITH CATG+1
    CASE VCAT='H'
      REPLACE CATH WITH CATH+1
    CASE VCAT='I'
      REPLACE CATI WITH CATI+1
    CASE VCAT='J'
      REPLACE CATJ WITH CATJ+1
    CASE VCAT='K'
      REPLACE CATK WITH CATK+1
    CASE VCAT='L'
      REPLACE CATL WITH CATL+1
    CASE VCAT='M'
      REPLACE CATM WITH CATM+1
    CASE VCAT='N'
      REPLACE CATN WITH CATN+1
    CASE VCAT='O'
      REPLACE CATO WITH CATO+1
    CASE VCAT='P'
      REPLACE CATP WITH CATP+1
    CASE VCAT='Q'
      REPLACE CATQ WITH CATQ+1
    CASE VCAT='R'
      REPLACE CATR WITH CATR+1
    CASE VCAT='S'
      REPLACE CATS WITH CATS+1
    CASE VCAT='T'
      REPLACE CATT WITH CATT+1
```

⋮

afficher la date de validité

Demander à l'utilisateur de valider

S'il valide, entrer un nouvel enregistrement LECTEUR.

Mettre à jour le compteur du nombre de lecteurs par catégorie

```
  ENDCASE

  CLOSE ALL

  SET CONSOLE OFF
  SET PRINT ON
  ?DATE(),'     Enregistrement lecteur.'
  DO AM_AFFLEC
  SET PRINT OFF
  SET CONSOLE ON
ENDIF

CLOSE ALL
CLEAR
RETURN
```

Imprimer les informations correspondant au nouvel enregistrement.
(grâce à un sous-programme d'affichage AM_AFFLEC)

```
*AM_SITULEC.PRG

PRIVATE REP

CLEAR
SET COLOR TO W
@1,28 SAY ´* SITUATION LECTEUR *´
SET COLOR TO BG
ACTIVATE WINDOW sitlec
@1,2 SAY ´1 - Recherche par nom´
@3,2 SAY ´2 - Recherche par code´
REP=0
@5,10 SAY ´Choix : ´ GET REP PICTURE ´#´
READ
DEACTIVATE WINDOW sitlec

DO CASE
CASE REP=1
  VNOM=SPACE(25)
  @4,5 SAY ´Nom       : ´ GET VNOM FUNCTION ´!´
  READ
  IF VNOM=SPACE(25)
    DO BEEP
    CLOSE ALL
    CLEAR
    RETURN
  ENDIF
  USE LECTEUR
  LOCATE FOR NOM=VNOM
  IF EOF()
    DO ERR_LEC
    RETURN
  ENDIF
  @5,5 SAY ´Liste     :´
  SCAN FOR NOM=VNOM
    ?´        ´
    ??NUMCAR,TRIM(NOM),TRIM(PRENOM),TRIM(RUE),TRIM(LIEU)
  ENDSCAN
  ?´ ´
  INPUT ´      Précisez le code lecteur : ´ TO VCAR
  USE LECTEUR ORDER numcar
  SEEK VCAR
  @4,5 CLEAR
  @3,5 SAY ´Code        : ´
  ??NUMCAR
  DO LECSITU

CASE REP=2
  USE LECTEUR
  CARTE=SPACE(3)
  @3,5 SAY ´Code        : ´ GET CARTE PICTURE ´###´
  READ
  VCAR=VAL(CARTE)
  IF VCAR=0
    DO BEEP
    CLOSE ALL
    CLEAR
    RETURN
  ENDIF
  USE LECTEUR ORDER numcar
  SEEK VCAR
  IF EOF()
    DO ERR_LEC
    RETURN
  ENDIF
  DO LECSITU
```

Proposer deux accès à la situation de lecteur
Lire le choix

Cas 1
Lire le nom

Si le nom n'existe pas, exécuter la procédure ERR_LEC (message d'erreur)

Afficher les coordonnées du ou des lecteurs ayant ce nom

Demander confirmation, ou indication du code lecteur souhaité.

Afficher ce code
passer à la procédure LECSITU
(établir la situation du lecteur)

Cas 2
Lire le code du lecteur

Si le code n'existe pas, exécuter la procédure ERR_LEC

Passer à la procédure LECSITU

```
ENDCASE

CLOSE ALL
CLEAR
RETURN

                      *PROCEDURES

Procedure ERR_LEC
@20,5 SAY 'Lecteur non enregistré ou erreur...>'
WAIT ' '
CLOSE ALL
CLEAR
RETURN
```

message d'erreur

```
Procedure LECSITU
PRIVATE VCAR,VCOD,TIT

@4,5 SAY 'Nom         : '
??TRIM(NOM),TRIM(PRENOM)
@5,5 SAY 'Catégorie   : '
??CATEG
@6,5 SAY 'Adresse     : '
??TRIM(RESID),TRIM(RUE)
@7,5 SAY '              '
??CODPOST,TRIM(LIEU)
```

Afficher les coordonnées

```
IF PENAL>=DATE()
  @8,5 SAY 'Pénalisé jusqu au : '
  ??PENAL
ENDIF
```

la pénalisation éventuelle

```
IF MESSG<>SPACE(160)
  @9,5 SAY 'Message     : '
  ??RTRIM(MESSG)
ENDIF
```

le message éventuel

```
SET COLOR TO W
@12,5 SAY 'Prêts : '
??NBPRET
@12,45 SAY 'Réservations : '
??NBRESER
VCAR=NUMCAR
```

le nombre des prêts du lecteur

le nombre des réservations du lecteur

```
USE PRET
SCAN FOR NUMCAR=VCAR
  ?'     '
  ??TRIM(CODBARR),DATESOR,'-',DATERET,'Renouv.',RENOU
ENDSCAN
```

Afficher le code document, les dates des prêts

```
USE RESERVAT
SCAN FOR NUMCAR=VCAR
  ?'                                              '
  ??CODBARR
ENDSCAN
```

Afficher le code document des réservations

```
SET COLOR TO BG
REP='0'
DO WHILE UPPER(REP)<>'N'
  VCOD=SPACE(7)
  @20,5 SAY 'Pour plus de précisions tapez le code document (sinon "Entrée") : '
  @20,69 GET VCOD FUNCTION '!'
  READ
  IF VCOD=SPACE(7)
    CLOSE ALL
    CLEAR
    RETURN
  ENDIF
```

DEBUT DE BOUCLE

Proposer des précisions sur les documents (prêts ou réservations)

Si elles ne sont pas demandées attendre un ↵ pour quitter

```
  USE DOCUMENT ORDER codbarr
  SEEK VCOD
  ACTIVATE WINDOW supple
  @2,2 SAY 'Cote  : '
  ??COTE
  @2,33 SAY 'N°invent.: '
  ??NUMINV
  @3,2 SAY 'Titre : '
  TIT=TITRE
  ??SUBSTR(TIT,1,39)
  @5,2 SAY 'Autre titre (O/N) ? ' GET REP
  READ
  DEACTIVATE WINDOW supple
ENDDO
@20,5 CLEAR TO 20,76
WAIT ' '
RETURN
```

Si elles sont demandées, afficher (dans une fenêtre) les coordonnées du document correspondant au code entré.

Proposer l'enchaînement sur un autre code.

FIN DE BOUCLE

Si l'utilisateur ne demande plus de code, attendre, quitter

```
*AM_MODLEC.PRG

PRIVATE REP

CLEAR
SET COLOR TO W
@1,28 SAY '* MODIFICATION LECTEUR *'
SET COLOR TO BG
 ACTIVATE WINDOW sitlec
 @1,2 SAY '1 - Entrée par le nom'
 @3,2 SAY '2 - Entrée par le code'
 REP=0
 @5,9 SAY 'Choix : ' GET REP PICTURE '#'
 READ
 DEACTIVATE WINDOW sitlec

 DO CASE
   CASE REP=1
     VNOM=SPACE(20)
     @4,5 SAY 'Nom      :' GET VNOM FUNCTION '!'
     READ
     USE LECTEUR
     LOCATE FOR NOM=VNOM
     IF EOF()
       DO MESS_ERR
       RETURN
     ENDIF
     @5,5 SAY 'Liste    : '
     SCAN FOR NOM=VNOM
       ?'     '
       ??NUMCAR,TRIM(NOM),TRIM(PRENOM),TRIM(RUE),TRIM(LIEU)
     ENDSCAN
     ?' '
     INPUT '      Précisez le code lecteur : ' TO VCAR
     LOCATE FOR NUMCAR=VCAR
     DO LECMOD

  CASE REP=2
    VCAR=0
    @4,5 SAY 'Code lecteur : ' GET VCAR PICTURE '###'
    READ
    USE LECTEUR  ORDER NUMCAR
    ~~LOCATE FOR NUMCAR=VCAR~~  SEEK VCAR
    IF EOF()
      DO MESS_ERR
      RETURN
    ENDIF
    DO LECMOD
ENDCASE

CLOSE ALL
CLEAR
RETURN

                    *PROCEDURES

Procedure MESS_ERR
@20,5 SAY 'Lecteur non enregistré...>'
DO BEEP
DO PAUSE
CLOSE ALL
CLEAR
RETURN


Procedure LECMOD
```

Proposer deux accès à la modification d'un enregistrement lecteur

Cas 1

Lire le nom

Si le nom n'existe pas exécuter MESS-ERR (message d'erreur)

S'il existe donner la liste des coordonnées du ou des lecteurs ayant ce nom

Demander confirmation, ou indication du code souhaité

Passer à la procédure LECMOD (pour exécuter les modifications)

Cas 2

Lire le code du lecteur

Si le nom n'existe pas, exécuter la procédure MESS-ERR.

message d'erreur

```
VPEN=PENAL
VCAT=CATEG
VNOM=NOM
VPRE=PRENOM
VRES=RESID
VRUE=RUE
VCOP=CODPOST
VLIE=LIEU
VNBP=NBPRET
VNBR=NBRESER
VMES=MESSG
VVAL=VALID
@4,5 CLEAR
@4,5  SAY ´Catégorie    : ´ GET VCAT FUNCTION ´!´
@5,5  SAY ´Nom          : ´ GET VNOM FUNCTION ´!´
@6,5  SAY ´Prénom       : ´ GET VPRE FUNCTION ´!´
@7,5  SAY ´Résidence    : ´ GET VRES
@8,5  SAY ´Rue          : ´ GET VRUE
@9,5  SAY ´Code postal  : ´ GET VCOP PICTURE ´#####´
@10,5 SAY ´Localité     : ´ GET VLIE FUNCTION ´!´
@11,5 SAY ´Observation  : ´ GET VMES
@12,5 SAY ´Validité     : ´ GET VVAL FUNCTION ´D´
@13,5 SAY ´Pénalisation : ´ GET VPEN FUNCTION ´D´
READ
REP=´N´
@20,5 SAY ´Validation (O/N) ? ´ GET REP
READ
IF UPPER(REP)=´O´
  REPLACE CATEG  WITH VCAT
  REPLACE NOM    WITH VNOM
  REPLACE PRENOM WITH VPRE
  REPLACE RESID  WITH VRES
  REPLACE RUE    WITH VRUE
  REPLACE LIEU   WITH VLIE
  REPLACE MESSG  WITH VMES
  REPLACE VALID  WITH VVAL
  REPLACE PENAL  WITH VPEN
  CLOSE ALL
  SET CONSOLE OFF
  SET PRINT ON
  ?DATE(),´     Modification lecteur.´
  ?´   Pénalisation : ´,VPEN
  DO AM_AFFLEC
  SET PRINT OFF
  SET CONSOLE ON
ENDIF
RETURN
```

Initialiser les variables

Lire les champs successifs

Demander à l'utilisateur s'il valide

Si oui, entrer les nouvelles valeurs dans l'enregistrement du lecteur

Imprimer les informations correspondant à l'enregistrement du lecteur, modifié.
(grâce au sous-programme d'affichage AM_AFFLEC)

```
*AM_ENRDOC.PRG

PRIVATE REP

CLEAR
SET COLOR TO W
@2,26 SAY '* ENREGISTREMENT DOCUMENT *'
SET COLOR TO BG

VTYP=0
VCOT=SPACE(10)
VINV=SPACE(8)
VAN=SPACE(10)
VTIT=SPACE(50)
VM1=SPACE(15)
VM2=SPACE(15)
VM3=SPACE(15)
VRES=0
VPR=0
@6,5 SAY 'Code barre    : ' GET VCOD PICTURE '######X' FUNCTION '!'
READ
IF VCOD=SPACE(7)
  CLOSE ALL
  CLEAR
  RETURN
ENDIF

USE DOCUMENT ORDER codbarr
SEEK VCOD
IF CODBARR=VCOD
  @8,5 SAY 'Titre : '+ TITRE
  @9,5 SAY 'Ce document est déjà enregistré...>'
  DO BEEP
  DO PAUSE
  CLOSE ALL
  CLEAR
  RETURN
ENDIF
CLOSE ALL

@7,5 SAY 'Type          : ' GET VTYP PICTURE '#'
@8,5 SAY 'Cote          : ' GET VCOT FUNCTION '!'
@9,5 SAY 'N° inventaire : ' GET VINV PICTURE '#####X##'
READ
DO WHILE VINV=SPACE(8)
  @10,5 SAY 'Le N°inventaire est obligatoire.'
  @9,22 GET VINV PICTURE '#####X##'
  READ
  @10,5 CLEAR
ENDDO
@10,5 SAY 'Année,N°      : ' GET VAN PICTURE '####X#####'
READ
DO WHILE VAN=SPACE(10)
  @11,5 SAY 'L année est obligatoire.'
  @10,22 GET VAN PICTURE '####X#####'
  READ
  @11,5 CLEAR
ENDDO
@11,5 SAY 'Titre         : ' GET VTIT FUNCTION '!'
READ
DO WHILE VTIT=SPACE(50)
  @12,5 SAY 'Le titre est obligatoire.'
  @11,22 GET VTIT FUNCTION '!'
  READ
  @12,5 CLEAR
ENDDO
```

Initialiser les variables

Lire le code du document
Si aucun code n'est entré, quitter

Vérifier si ce document a déjà été enregistré ; afficher son titre, le cas échéant, attendre, quitter

Lire les coordonnées suivantes
Contrôler l'entrée obligatoire du n° inventaire

de l'année, n°

du titre

```
@12,5 SAY '1er mot-titre : ' GET VM1 FUNCTION '!'
@13,5 SAY '2e  mot-titre : ' GET VM2 FUNCTION '!'
@14,5 SAY '3e  mot-titre : ' GET VM3 FUNCTION '!'
READ
REP='N'
@16,5 SAY 'Validation (O/N) ? ' GET REP
READ
IF UPPER(REP)='O'
  USE DOCUMENT
  APPEND BLANK
  REPLACE TYPDOC WITH VTYP
  REPLACE CODBARR WITH VCOD
  REPLACE COTE WITH VCOT
  REPLACE NUMINV WITH VINV
  REPLACE ANNEENO WITH VAN
  REPLACE TITRE WITH VTIT
  REPLACE MOT1 WITH VM1
  REPLACE MOT2 WITH VM2
  REPLACE MOT3 WITH VM3
  CLOSE ALL

  SET CONSOLE OFF
  SET PRINT ON
  ?DATE(),'     Enregistrement document.'
  DO AM_AFFDOC
  SET PRINT OFF
  SET CONSOLE ON
ENDIF

CLOSE ALL
CLEAR
RETURN
```

Lire les coordonnées suivantes

Demander à l'utilisateur s'il valide

Si oui, entrer un nouvel enregistrement DOCUMENT

Imprimer les informations correspondant au nouvel enregistrement. (utiliser le sous-programme d'affichage AM_AFFDOC)

AM_SITUDOC.PRG

```
*AM_SITUDOC.PRG

PRIVATE REP,VINV,VCOD,VCOT,TIT

CLEAR
SET COLOR TO W
@1,29 SAY '* SITUATION DOCUMENT *'
SET COLOR TO BG
ACTIVATE WINDOW sitdoc

@1,1 SAY '1 - Recherche par N°inventaire'
@3,1 SAY '2 - Recherche par code document'
@5,1 SAY '3 - Recherche par cote'
@7,1 SAY '4 - Recherche par mot du titre'

REP=0
@9,10 SAY 'Choix : ' GET REP PICTURE '#'
READ
DEACTIVATE WINDOW sitdoc

DO CASE
  CASE REP=1
    VINV=SPACE(8)
    @4,5 SAY 'N° inventaire : ' GET VINV PICTURE '#####X##'
    READ
    IF VINV=SPACE(8)
      DO BEEP
      CLOSE ALL
      CLEAR
      RETURN
    ENDIF
    USE DOCUMENT ORDER numinv
    SEEK VINV
    IF EOF()
      DO DOC_ERR
      RETURN
    ENDIF
    VCOD=CODBARR
    DO DOCSITU
  CASE REP=2
    VCOD=SPACE(7)
    @4,5 SAY 'Code document : ' GET VCOD PICTURE '999999X' FUNCTION '!'
    READ
    IF VCOD=SPACE(7)
      DO BEEP
      CLOSE ALL
      CLEAR
      RETURN
    ENDIF
    USE DOCUMENT ORDER codbarr
    SEEK VCOD
    IF EOF()
      DO DOC_ERR
      RETURN
    ENDIF
    DO DOCSITU
  CASE REP=3
    VCOT=SPACE(10)
    @4,5 SAY 'Cote  : ' GET VCOT FUNCTION '!'
    READ
    IF VCOT=SPACE(10)
      DO BEEP
      CLOSE ALL
      CLEAR
      RETURN
    ENDIF
```

Proposer l'accès à la situation document

Lire le choix

Cas 1

Lire le n° inventaire

Si le n° inventaire n'existe pas, exécuter DOC-ERR (message d'erreur)

exécuter la procédure DOCSITU (établir la situation du document)

Cas 2

Lire le code du document

Vérifier qu'il existe

exécuter la procédure DOCSITU

Cas 3

Lire la cote du document

```
  USE DOCUMENT ORDER cote
  SEEK VCOT
  IF EOF()
    DO DOC_ERR
    RETURN
  ENDIF
  @7,5 SAY 'Liste :'
  SCAN FOR COTE=VCOT
    TIT=TITRE
    ?'  '
    ??COTE,NUMINV,SUBSTR(TIT,1,30)
  ENDSCAN
  ?' '
  ACCEPT '     Entrez le N° inventaire voulu : ' TO VINV
  USE DOCUMENT ORDER numinv
  SEEK VINV
  VCOD=CODBARR
  @7,5 CLEAR
  DO DOCSITU
CASE REP=4
  STORE SPACE(20) TO MCL1,MCL2,MCL3
  @4,5 SAY 'Premier mot    : ' GET MCL1 FUNCTION '!'
  @5,5 SAY 'Deuxième mot   : ' GET MCL2 FUNCTION '!'
  @6,5 SAY 'Troisième mot  : ' GET MCL3 FUNCTION '!'
  READ
  IF MCL1=SPACE(20)
    DO BEEP
    CLOSE ALL
    CLEAR
    RETURN
  ELSE
    IF MCL2=SPACE(20) .AND. MCL3=SPACE(20)
      USE DOCUMENT
      @8,5
      SCAN FOR MOT1=MCL1 .OR. MOT2=MCL1 .OR. MOT3=MCL1
        DO AFFICH
      ENDSCAN
      WAIT ' '
      CLOSE ALL
      CLEAR
      RETURN
    ELSE
      IF MCL3=SPACE(20)
        USE DOCUMENT
        @8,5
        SCAN FOR MOT1=MCL1 .OR. MOT2=MCL1 .OR. MOT3=MCL1
          IF MOT1=MCL2 .OR. MOT2=MCL2 .OR. MOT3=MCL2
            DO AFFICH
          ENDIF
        ENDSCAN
        WAIT ' '
        CLOSE ALL
        CLEAR
        RETURN
      ELSE
        DO RECH_MOT WITH MCL1,MCL2,MCL3
        DO RECH_MOT WITH MCL2,MCL1,MCL3
        DO RECH_MOT WITH MCL3,MCL1,MCL2
      ENDIF
    ENDIF
  ENDIF
ENDCASE

CLOSE ALL
CLEAR
RETURN
```

Vérifier qu'elle existe

Afficher les coordonnées du ou des documents correspondant à cette cote

Demander confirmation ; ou indication du n° inventaire souhaité

Exécuter la procédure DOCSITU

Cas 4

Lire les 3 mots entrés

Si le premier mot n'est pas entré émettre un signal sonore, quitter.

Sinon
si les 2 mots suivants ne sont pas entrés :
exécuter la procédure AFFICH pour les documents dont le titre contient le mot-titre entré.

si le 3e mot n'est pas entré exécuter la procédure AFFICH pour les documents dont le titre contient les deux mots-titre entrés

Sinon, exécuter la procédure RECH-MOT, à 3 reprises :
en plaçant successivement en 1ère position le 1er mot-titre entré
le 2e mot-titre entré
le 3e mot-titre entré

*PROCEDURES

```
Procedure RECH_MOT
PARAMETERS M1,M2,M3
```

Passage de paramètres

```
USE DOCUMENT
@8,5
SCAN FOR MOT1=M1
  IF MOT2=M2 .AND. MOT3=M3 .OR. MOT2=M3 .AND. MOT3=M2
    DO AFFICH
  ENDIF
ENDSCAN
WAIT ' '
CLOSE ALL
RETURN
```

Rechercher la position de M1 en tant que 1er mot-titre
Si M1 et M2 sont trouvés en position de 2e ou 3e mot-titre
exécuter la procédure AFFICH

```
Procedure AFFICH
~~TIT=TITRE~~
?'     '
??NUMINV,COTE,~~SUBSTR(TIT,1,50)~~ TITRE
RETURN
```

Afficher le n° d'inventaire, la cote, le titre du document correspondant au(x) mot(s) entré(s).

```
Procedure DOC_ERR
  @20,5 SAY 'Document non enregistré...>'
  DO BEEP
  DO PAUSE
  CLOSE ALL
  CLEAR
RETURN
```

Afficher un message d'erreur

```
Procedure DOCSITU
@5,5 SAY 'Titre : '
??TITRE
```

Afficher le titre du document

```
SET COLOR TO W
USE PRET
LOCATE FOR CODBARR=VCOD
IF EOF()
  @8,5 SAY 'Document présent.'
ELSE
  @8,5 SAY 'En prêt :'
  ?'     '
  ??NUMCAR,' ',DATESOR,'-',DATERET,'Renouv.',RENOU
ENDIF
```

Indiquer s'il est en prêt ou s'il est disponible

les informations éventuelles concernant son prêt.

```
USE RESERVAT
LOCATE FOR CODBARR=VCOD
IF EOF()
  @8,45 SAY 'Non réservé.'
ELSE
  @8,45 SAY 'Réservé par :'
  SCAN FOR CODBARR=VCOD
    ?'                                   '
    ??NUMCAR
  ENDSCAN
ENDIF
~~WAIT ' '~~
~~CLOSE ALL~~
~~CLEAR~~
~~RETURN~~

SET COLOR TO BG
REP='O'
```

Indiquer s'il est non réservé
s'il est réservé
et par quel(s) lecteur(s) en affichant le(s) code(s) lecteur

DEBUT DE BOUCLE

```
DO WHILE UPPER(REP)<>'N'
  @20,5 SAY 'Pour plus de précisions tapez le code du lecteur (sinon "Entrée") :'
  CARTE=SPACE(3)
  @20,71 GET CARTE PICTURE '###'
  READ
  VCAR=VAL(CARTE)
  IF VCAR=0
    CLOSE ALL
    CLEAR
    RETURN
  ENDIF
  USE LECTEUR ORDER numcar
  SEEK VCAR
  ACTIVATE WINDOW supple
  @2,2 SAY 'Nom : '
  ??TRIM(NOM),TRIM(PRENOM)
  @3,2 SAY 'Catégorie : '
  ??CATEG
  @5,2 SAY 'Autre lecteur (O/N) ? ' GET REP
  READ
  DEACTIVATE WINDOW supple
ENDDO
@20,5 CLEAR TO 20,76
WAIT ' '
RETURN
```

Proposer des précisions sur le(s) lecteur(s) ayant emprunté ou réservé ce document

Si elles sont demandées, afficher (dans une fenêtre) les coordonnées du lecteur correspondant au code entré.

FIN DE BOUCLE

Effacer la question si l'utilisateur ne demande plus de code lecteur attendre, quitter

```
*AM_MODDOC.PRG

PRIVATE REP,TIT

CLEAR
SET COLOR TO W
@1,27 SAY ´* MODIFICATION DOCUMENT *´
SET COLOR TO BG

ACTIVATE WINDOW sitdoc
@1,2 SAY ´1 - Entrée par N°inventaire´
@3,2 SAY ´2 - Entrée par code document´
@5,2 SAY ´3 - Entrée par cote´
REP=0
@8,10 SAY ´Choix : ´ GET REP PICTURE ´#´
READ
DEACTIVATE WINDOW sitdoc

DO CASE
  CASE REP=1
    VINV=SPACE(8)
    @4,5 SAY ´N° inventaire :´ GET VINV PICTURE ´#####X##´
    READ
    USE DOCUMENT ORDER numinv
    SEEK VINV
    IF EOF()
      DO ERREUR
      RETURN
    ENDIF
    DO DOCMOD

  CASE REP=2
    VCOD=SPACE(7)
    @4,5 SAY ´Code document :´ GET VCOD FUNCTION ´!´
    READ
    USE DOCUMENT ORDER codbarr
    SEEK VCOD
    IF EOF()
      DO ERREUR
      RETURN
    ENDIF
    DO DOCMOD

  CASE REP=3
    VCOT=SPACE(10)
    @4,5 SAY ´Cote :´ GET VCOT FUNCTION ´!´
    READ
    USE DOCUMENT ORDER cote
    SEEK VCOT
    IF EOF()
      DO ERREUR
      RETURN
    ENDIF
    @6,5 SAY ´Liste :´
    SCAN FOR COTE=VCOT
      TIT=TITRE
      ?´   ´
      ??COTE,´-´,NUMINV,´-´,SUBSTR(TIT,1,30)
    ENDSCAN
    ?´ ´
    INPUT ´      Entrez le N° inventaire voulu : ´ TO VINV
    USE DOCUMENT ORDER numinv
    SEEK VINV
    @6,5 CLEAR
    DO DOCMOD
ENDCASE
```

Proposer 3 accès à la modification d'un document enregistré.

Cas 1

Lire le n° inventaire

Si le n° inventaire n'existe pas exécuter la procédure ERREUR

exécuter la procédure DOCMOD

Cas 2

Lire le code du document

⋮

id.

Cas 3

Lire la cote du document

⋮

id.

Affiche les coordonnées du ou des documents correspondant à cette cote

Demander confirmation ; ou indication du document souhaité, par l'entrée de son n° inventaire.

⋮

id.

```
CLOSE ALL
CLEAR
RETURN

                              *PROCEDURES

Procedure ERREUR
@20,5 SAY 'Document non enregistré...>'
DO BEEP
DO PAUSE
CLOSE ALL
CLEAR
RETURN


Procedure DOCMOD
VCOD=CODBARR
VTYP=TYPDOC
VCOT=COTE
VINV=NUMINV
VTIT=TITRE
VAN =ANNEENO
VM1 =MOT1
VM2 =MOT2
VM3 =MOT3
VRES=RESER
VPR =PRET
@4,5 CLEAR
@4,5  SAY 'Type          : ' GET VTYP FUNCTION '!'
@5,5  SAY 'Cote          : ' GET VCOT FUNCTION '!'
@6,5  SAY 'N°inventaire  : ' GET VINV PICTURE '#####X##'
@7,5  SAY 'Année,N°      : ' GET VAN PICTURE '####X#####'
@8,5  SAY 'Titre         : ' GET VTIT FUNCTION '!'
@9,5  SAY '1er mot-titre : ' GET VM1 FUNCTION '!'
@10,5 SAY '2e  mot-titre : ' GET VM2 FUNCTION '!'
@11,5 SAY '3e mot-titre  : ' GET VM3 FUNCTION '!'
READ
REP='N'
@20,5 SAY 'Validation (O/N) ? ' GET REP
READ
IF UPPER(REP)='O'
  REPLACE TYPDOC  WITH VTYP
  REPLACE COTE    WITH VCOT
  REPLACE NUMINV  WITH VINV
  REPLACE ANNEENO WITH VAN
  REPLACE TITRE   WITH VTIT
  REPLACE MOT1    WITH VM1
  REPLACE MOT2    WITH VM2
  REPLACE MOT3    WITH VM3
  CLOSE ALL
  SET CONSOLE OFF
  SET PRINT ON
  ?DATE(),'     Modification document.'
  DO AM_AFFDOC
  SET PRINT OFF
  SET CONSOLE ON
ENDIF
RETURN
```

Afficher un message d'erreur attendre, quitter

Initialiser les variables

Lire les valeurs entrées

Demander à l'utilisateur s'il valide
Si oui, remplacer les valeurs dans l'enregistrement du document

Imprimer les informations correspondant à l'enregistrement modifié du document
(en utilisant le sous-programme d'affichage AM_AFFDOC)

```
*AM_RELAN.PRG

PRIVATE N

CLEAR
SET COLOR TO W
@1,28 SAY '* RELANCES *'

SET COLOR TO BG
ACTIVATE WINDOW affstat
N=1
EJECT PAGE
DO WHILE N<4
  DO RAPPEL WITH N
  N=N+1
ENDDO
DEACTIVATE WINDOW affstat
SET PRINT ON
?DATE(),'    Relances terminées.'
?' '
?'Liste des 3e relances non abouties : '
?' '
SET PRINT OFF
USE PRET
LIST FIELDS NUMCAR, CODBARR,DATERET,RELAN OFF ALL FOR RELAN=3 .AND. DATERET<(DATE()-((INTREL*3)+2)) TO PRINTER
```

Ouvrir une fenêtre (pour conserver l'intitulé "RELANCES")

Se placer en début de page

DEBUT DE BOUCLE

Exécuter l'ensemble des relances pour la valeur N avec la procédure RAPPEL

incrémenter N

FIN DE BOUCLE

Fermer la fenêtre

Imprimer date et message de fin de relances

Imprimer la liste des 3e relances n'ayant pas abouti.

```
CLOSE ALL
CLEAR
RETURN


                         *PROCEDURES


Procedure RAPPEL
PARAMETERS N
USE PRET IN 1
SCAN FOR DATERET<(DATE()-(INTREL*N))
  IF RELAN<N
    DO LETTRE WITH N,NUMCAR,CODBARR,DATERET
    SELECT 1
    REPLACE RELAN WITH N
  ENDIF
ENDSCAN
USE IN 1
RETURN
```

Dans le fichier PRET

DEBUT DE BOUCLE

Chercher les prêts dont la date de retour est dépassée

Lire le champ 'nombre de relances'

imprimer la lettre de relance si le nombre de relances est inférieur à N (avec la procédure LETTRE)

mettre à jour le nombre de relances de l'enregistrement du prêt

FIN DE BOUCLE

```
Procedure LETTRE
PARAMETERS N,CARTE,CODE,DATELIM
PRIVATE VNOM,VPRE,VRES,VRUE,VCOP,VLIE,VMES,VCOT,VTIT,VINV

USE LECTEUR IN 2 ORDER numcar
SELECT 2
SEEK CARTE
IF EOF()
  DO ERR_CAR
  USE IN 2
  RETURN
ELSE
  VNOM=NOM
  VPRE=PRENOM
  VRES=RESID
  VRUE=RUE
  VCOP=CODPOST
  VLIE=LIEU
```

Passage de paramètres.

Vérifier que le code lecteur indiqué dans le fichier PRET, existe dans le fichier LECTEUR.

S'il existe initialiser les variables en leur affectant les valeurs des champs de l'enregistrement du lecteur

```
VMES=MESSG

USE DOCUMENT IN 2 ORDER codbarr
SEEK CODE
IF EOF()
  DO ERR_CODE
  USE IN 2
  RETURN
ELSE
  VCOT=COTE
  VTIT=TITRE
  VINV=NUMINV
  USE IN 2

  @1,2 SAY 'Lecteur  : '
  ??TRIM(VNOM),TRIM(VPRE)
  @3,2 SAY 'Document : '
  ?'      ',VINV,VCOT,VTIT
  @5,2 SAY 'Message (entrez ou modifiez) : '
  @7,2 GET VMES
  READ
  USE LECTEUR IN 2 ORDER numcar
  SEEK CARTE
  REPLACE MESSG WITH VMES
  USE IN 2

  SET CONSOLE OFF
  SET PRINT ON
  ?'E.N.S.B.- BIBLIOTHEQUE              Villeurbanne, le ',DATE()
  ?'17/21 Bv du 11 Novembre 1918'
  ?'69623 VILLEURBANNE CEDEX'
  ?' '
  ?'Tél. 78 89 64 45                    à'
  ?' '
  ?' '
  ?' '
  ?'                                   ',TRIM(VNOM),VPRE
  ?'                                   ',VRES
  ?'                                   ',VRUE
  ?'                                   ',VCOP,VLIE

  DO CASE
    CASE N=1
    ?' '
    ?' '
    ?' '
    ?'1er RAPPEL'
    ?'----------'
    ?' '
    ?'Veuillez rapporter le document ci-dessous que vous deviez'
    ?'rendre le : ',DATELIM
    CASE N=2
    ?' '
    ?' '
    ?' '
    ?'2e  RAPPEL'
    ?'----------'
    ?' '
    ?'Veuillez rapporter le document ci-dessous que vous deviez'
    ?'rendre le : ',DATELIM
    CASE N=3
    ?' '
    ?' '
    ?' '
    ?'3e  RAPPEL'
```

Annotations manuscrites :

Vérifier que le code document indiqué dans le fichier PRET existe dans le fichier DOCUMENT

id.

Affiche les coordonnées du lecteur

Lire le message éventuellement entré par l'utilisateur

Début d'impression

Pour chaque lettre : imprimer l'en-tête, la date

le nom et l'adresse du lecteur.

Selon la valeur de N, imprimer

Cas 1

Cas 2 (ici, le texte est identique à celui du cas 1, il pourra subir des modifications si besoin est)

ou Cas 3

```
    ?'----------'
    ?' '
    ?'Veuillez rapporter le document ci-dessous que vous deviez'
    ?'rendre le : ',DATELIM
    ?' '
    ?'Nous vous prions de régulariser votre situation dans un'
    ?'délai de 48 h. En cas d empêchement veuillez nous téléphoner.'
    ?'En l absence de réponse de votre part, nous serions au regret'
    ?'de porter le litige devant les services administratifs de'
    ?'l Ecole.'
   ENDCASE
   ?' '
   ?'Cote  : ',VCOT
   ?'Titre : ',VTIT
   ?'N°inv.: ',VINV
   ?' '
   ?'OBSERVATIONS : ',VMES
   EJECT PAGE
   SET PRINTER OFF
   SET CONSOLE ON
  ENDIF
ENDIF
RETURN


Procedure ERR_CAR
@20,5 SAY 'Le lecteur N° '+ CARTE
@21,5 SAY 'ne se trouve pas dans le fichier LECTEUR...>'
DO BEEP
DO PAUSE
RETURN


Procedure ERR_CODE
@20,5 SAY 'Le document N° ' + CODE
@21,5 SAY 'ne se trouve pas dans le fichier DOCUMENT'
DO BEEP
DO PAUSE
RETURN
```

Pour chaque lettre, imprimer les coordonnées du document non-retourné et les observations entrées par l'utilisateur

Faire le saut de page

Fin d'impression

Signaler l'incohérence entre les données du fichier PRET et celles du fichier LECTEUR

id PRET

DOCUMENT

(ces deux erreurs ne devraient pas se produire en cas de bonne marche du programme...)

```
*AM_EDIT.PRG

PRIVATE REP

CLEAR
SET COLOR TO W
@1,32 SAY '* EDITIONS *'
SET COLOR TO BG

ACTIVATE WINDOW sitdoc
REP=0
@1,2 SAY '1 - Liste des prêts'
@3,2 SAY '2 - Liste des retards'
@5,2 SAY '3 - Liste des non-réinscrits'
@7,8 SAY 'Choix : ' GET REP PICTURE '#'
READ
DEACTIVATE WINDOW sitdoc

DO CASE
  CASE REP=1
    CLEAR
    SET COLOR TO W
    @1,25 SAY '* Edition des prêts *'
    SET COLOR TO BG
    DO AM_EDIPRET
  CASE REP=2
    CLEAR
    SET COLOR TO W
    @1,23 SAY '* Edition des retards *'
    SET COLOR TO BG
    DO AM_EDIRET
  CASE REP=3
    CLEAR
    SET COLOR TO W
    @1,18 SAY '* Liste des lecteurs non réinscrits *'
    SET COLOR TO BG
    DO AM_EDILECT
ENDCASE

CLOSE ALL
CLEAR
RETURN
```

AM_EDIT.PRG

Proposer les différents choix

Cas 1

Exécuter le sous-programme AM_EDIPRET

Cas 2

Exécuter le sous-programme AM_EDIRET

Cas 3

Exécuter le sous-programme AM_EDILECT

AM_EDIPRET.PRG

```
* AM_EDIPRET.PRG

@4,10 SAY 'Appuyez sur une touche quand imprimante prête...>'
WAIT ' '
SET FIELDS TO
SELECT 2
USE PRET.DBF AGAIN NOUPDATE
USE DOCUMENT.DBF AGAIN NOUPDATE IN 1 ORDER CODBARR
SET EXACT ON
SET FILTER TO FOUND(1)
SET RELATION TO B->CODBARR INTO A
SET SKIP TO A
QBE__SAFE = SET("SAFETY")
QBE__CATA = SET("CATALOG")
SET SAFETY OFF
SET CATALOG OFF
GO TOP
SET FIELDS TO A->COTE,A->NUMINV,A->TITRE,B->DATERET,A->CODBARR
QBE__21 = STR(RAND(-1)*100000000,8)
COPY TO &QBE__21
USE &QBE__21 NOSAVE NOUPDATE
IF RECCOUNT() > 1
QBE__22 = STR(RAND(-1)*100000000,8)
SORT TO &QBE__22 ON DATERET/A
USE &QBE__22 NOSAVE NOUPDATE
ENDIF
SET SAFETY &QBE__SAFE
SET CATALOG &QBE__CATA
SET FIELDS TO
SET FIELDS TO COTE,NUMINV,TITRE,DATERET,CODBARR
GO TOP
SET CONSOLE OFF
LIST OFF TO PRINTER
@20,10 SAY 'Impression terminée...>'
WAIT ' '
SET CONSOLE ON

CLOSE ALL
CLEAR
RETURN
```

partie réalisée avec le générateur d'applications

mise en relation des fichiers PRET et DOCUMENT pour afficher pour chaque prêt :
cote, n° inventaire, titre, date de retour, code du document
(dans l'ordre des dates de retour)

```
*AM_EDIRET.PRG

@4,10 SAY 'Appuyez sur une touche quand imprimante prête...>'
WAIT ' '
SET FIELDS TO
SELECT 1
SELECT 3
USE PRET.DBF AGAIN
QBE__CT = 1
DO WHILE LEN(TAG(QBE__CT)) <> 0
IF TAG(QBE__CT) = "CODBARR"
EXIT
ELSE
QBE__CT = QBE__CT+1
ENDIF
ENDDO
IF LEN(TAG(QBE__CT)) <> 0
SET ORDER TO TAG CODBARR
ELSE
INDEX ON CODBARR TAG CODBARR
ENDIF
RELEASE QBE__CT
SELECT 1
USE DOCUMENT.DBF AGAIN NOUPDATE
USE PRET.DBF AGAIN NOUPDATE IN 3 ORDER CODBARR
USE LECTEUR.DBF AGAIN NOUPDATE IN 2 ORDER NUMCAR
SET EXACT ON
SET FILTER TO FOUND(3) .AND. ((C->DATERET< DATE())) .AND. FOUND(2)
SET RELATION TO A->CODBARR INTO C
SELECT 3
SET RELATION TO C->NUMCAR INTO B
SELECT 1
SET SKIP TO C,B
QBE__SAFE = SET("SAFETY")
QBE__CATA = SET("CATALOG")
SET SAFETY OFF
SET CATALOG OFF
GO TOP
SET FIELDS TO A->COTE,A->NUMINV,B->NOM,C->DATERET,C->RELAN
QBE__11 = STR(RAND(-1)*100000000,8)
COPY TO &QBE__11
USE &QBE__11 NOSAVE NOUPDATE
IF RECCOUNT() > 1
QBE__12 = STR(RAND(-1)*100000000,8)
SORT TO &QBE__12 ON DATERET/A
USE &QBE__12 NOSAVE NOUPDATE
ENDIF
SET SAFETY &QBE__SAFE
SET CATALOG &QBE__CATA
SET FIELDS TO
SET FIELDS TO COTE,NUMINV,NOM,DATERET,RELAN
GO TOP
SET CONSOLE OFF
LIST OFF TO PRINTER
@20,10 SAY 'Impression terminée...>'
WAIT ' '
SET CONSOLE ON

CLOSE ALL
CLEAR
RETURN
```

AM_EDIRET.PRG

Partie réalisée avec le générateur d'application

Mise en relation des fichiers PRET, DOCUMENT, LECTEUR

afficher dans l'ordre des dates de retour, pour chaque retard : Cote, n° inventaire, nom, date de retour nombre de relances.

```
*AM_EDILECT.PRG

@4,5
SET PRINT ON
?DATE(),'    Lecteurs non-réinscrits'
?' '
SET PRINT OFF
USE LECTEUR ORDER valid
LIST FIELDS NUMCAR,CATEG,NOM,NBPRET,VALID OFF ALL FOR VALID<DATE() TO PRINTER
@20,5
WAIT 'Impression terminée...>'
RETURN
```

AM_EDILEC.PRG

Lister pour chaque lecteur non-réinscrit :

son code, sa catégorie, son nom, son nombre de prêts éventuels.

```
*AM_STAT.PRG

PRIVATE REP

CLEAR
SET COLOR TO W
@1,30 SAY '* STATISTIQUES *'

SET COLOR TO BG
ACTIVATE WINDOW statis
@1,3 SAY '1 - Prêts par document'
@3,3 SAY '2 - Prêts par secteur/catégorie'
@5,3 SAY '3 - Total des prêts'
@7,3 SAY '4 - Tableau par secteur/catégorie'
@9,3 SAY '5 - Etat documents et lecteurs'
REP=0
@11,13 SAY 'Choix : ' GET REP PICTURE '#'
READ
DEACTIVATE WINDOW statis

DO CASE
  CASE REP=1
    DO AM_STAT1
  CASE REP=2
    DO AM_STAT2
  CASE REP=3
    DO AM_STAT3
  CASE REP=4
    DO AM_STAT4
  CASE REP=5
    DO AM_STAT5
ENDCASE

CLEAR
RETURN
```

AM_STAT.PRG

Proposer les différents choix

Lire le choix

Cas 1
exécuter le sous-programme AM_STAT 1
Cas 2 . 2
Cas 3 . 3
Cas 4 . 4
Cas 5 id 5

AM_STAT1.PRG

```
*AM_STAT1.PRG

PRIVATE VCOD,AN,REP

CLEAR
SET COLOR TO W
@1,28 SAY '* Prêts par document *'

SET COLOR TO BG
REP='O'
DO WHILE UPPER(REP)<>'N'
  @4,5 CLEAR
  VCOD=SPACE(7)
  @4,5 SAY 'Entrez le code du document : ' GET VCOD PICTURE '999999X' FUNCTION '!'
  READ
  IF VCOD=SPACE(7)
    CLOSE ALL
    CLEAR
    RETURN
  ENDIF
  USE DOCUMENT ORDER codbarr
  SEEK VCOD
  IF EOF()
    @20,5 SAY 'Document non enregistré...>'
    DO BEEP
    DO PAUSE
  ELSE
    AN=YEAR(DATE())
    @4,5
    SET PRINT ON
    ?'   ',DATE(),'   Prêts des 5 dernières années :'
    ?' '
    ?'                 Code   : ',VCOD
    ?'                 Cote   : ',COTE
    ?'           ',AN,'  : ',AN1
    ?'           ',AN-1,'  : ',AN2
    ?'           ',AN-2,'  : ',AN3
    ?'           ',AN-3,'  : ',AN4
    ?'           ',AN-4,'  : ',AN5
    ?'                *'
    ?' '
    SET PRINT OFF
    CLOSE ALL
  ENDIF
  @20,5 SAY 'Autre document (O/N) ? ' GET REP
  READ
ENDDO

CLOSE ALL
CLEAR
RETURN
```

DEBUT DE BOUCLE

Lire le code du document

Vérifier s'il existe

Afficher et imprimer
- Code
- cote
- nombre de prêts des 5 dernières années

Proposer l'enchaînement sur un autre document

FIN DE BOUCLE

```
*AM_STAT2.PRG

PRIVATE VSEC,VCAT,VNBCAT,AN,ANNEE,REP

CLEAR
SET COLOR TO W
@1,25 SAY '* Prêts par secteur/catégorie *'

SET COLOR TO BG
REP='0'
DO WHILE UPPER(REP)<>'N'
  @4,5 CLEAR
  VSEC=SPACE(2)
  VCAT=SPACE(1)
  ANNEE=SPACE(4)

  @4,5 SAY 'Année                  : ' GET ANNEE PICTURE '####'
  @5,5 SAY 'Secteur documentaire : ' GET VSEC
  @6,5 SAY 'Catégorie de lecteur : ' GET VCAT
  READ
  AN=VAL(ANNEE)
  IF AN=0
    CLOSE ALL
    CLEAR
    RETURN
  ENDIF
  IF YEAR(DATE())=AN
    USE SECCAT
  ELSE
    AN=SUBSTR(ANNEE,3)
    USE ('SECCAT'+AN)
  ENDIF
  DO CASE
    CASE VSEC='P'
      GO 1
    CASE VSEC='A'
      GO 2
    CASE VSEC='B'
      GO 3
    CASE VSEC='C'
      GO 4
    CASE VSEC='AB'
      GO 5
    CASE VSEC='BA'
      GO 6
    CASE VSEC='BB'
      GO 7
    CASE VSEC='BC'
      GO 8
    CASE VSEC='CA'
      GO 9
    CASE VSEC='W'
      GO 10
    CASE VSEC='X'
      GO 11
    CASE VSEC='Y'
      GO 12
    CASE VSEC='Z'
      GO 13
    OTHERWISE
      @20,5 SAY 'Le secteur '+VSEC+'n est pas prévu...>'
      DO BEEP
      DO PAUSE
      CLOSE ALL
      CLEAR
      RETURN
```

DEBUT DE BOUCLE
Initialiser les variables

Lire les données entrées par l'utilisateur

Ouvrir le fichier SECCAT*.dbf qui correspond à l'année demandée

Selon le secteur documentaire voulu aller à l'enregistrement correspondant

Indiquer si le secteur demandé n'existe pas, attendre, quitter.

```
ENDCASE

DO CASE
  CASE VCAT='A'
    VNBCAT=CATA
  CASE VCAT='B'
    VNBCAT=CATB
  CASE VCAT='C'
    VNBCAT=CATC
  CASE VCAT='D'
    VNBCAT=CATD
  CASE VCAT='E'
    VNBCAT=CATE
  CASE VCAT='F'
    VNBCAT=CATF
  CASE VCAT='G'
    VNBCAT=CATG
  CASE VCAT='H'
    VNBCAT=CATH
  CASE VCAT='I'
    VNBCAT=CATI
  CASE VCAT='J'
    VNBCAT=CATJ
  CASE VCAT='K'
    VNBCAT=CATK
  CASE VCAT='L'
    VNBCAT=CATL
  CASE VCAT='M'
    VNBCAT=CATM
  CASE VCAT='N'
    VNBCAT=CATN
  CASE VCAT='O'
    VNBCAT=CATO
  CASE VCAT='P'
    VNBCAT=CATP
  CASE VCAT='Q'
    VNBCAT=CATQ
  CASE VCAT='R'
    VNBCAT=CATR
  CASE VCAT='S'
    VNBCAT=CATS
  CASE VCAT='T'
    VNBCAT=CATT
  OTHERWISE
    @20,5 SAY 'La catégorie '+VCAT+'n est pas prévue...>'
    DO BEEP
    DO PAUSE
    CLOSE ALL
    CLEAR
    RETURN
ENDCASE
CLOSE ALL
@4,5 CLEAR
SET PRINT ON
?'              Année                  : ',ANNEE
?'              Secteur documentaire : ',VSEC
?'              Catégorie de lecteur : ',VCAT
?'              Nombre de prêts        : ',VNBCAT
?'              *'
?' '
SET PRINT OFF
@15,5 SAY 'Autre recherche (O/N) ? ' GET REP
READ
ENDDO
CLEAR
RETURN
```

Selon la catégorie demandée affiche le nombre total de prêts correspondant.

Indiquer si la catégorie n'existe pas, attendre, quitter.

Afficher et imprimer les données trouvées.

Proposer l'enchaînement sur une autre recherche

FIN DE BOUCLE

```
*AM_STAT3.PRG

PRIVATE AN,ANNEE,REP,TOTAN

CLEAR
SET COLOR TO W
@1,28 SAY '* Total des prêts *'

SET COLOR TO BG
REP='O'
DO WHILE UPPER(REP)<>'N'
  @4,5 CLEAR
  ANNEE=SPACE(4)
  @4,5 SAY 'Année souhaitée : ' GET ANNEE PICTURE '####'
  READ
  AN=VAL(ANNEE)
  IF AN=0
    CLOSE ALL
    CLEAR
    RETURN
  ENDIF

  USE COMPTEUR ORDER année
  SEEK AN
  IF EOF()
    @20,5 SAY 'Cette année n est pas dans le fichier...>'
    DO BEEP
    DO PAUSE
    CLOSE ALL
    CLEAR
    RETURN
  ELSE
    TOTAN=JAN+FEV+MAR+AVR+MAI+JUN+JUL+AOU+SEP+OCT+NOV+DEC
    @4,5
    *SET PRINT ON
    ?'          ',AN,'    Total des prêts : ',TOTAN
    ?' '
    ?'             Janvier : ',JAN,'     Juillet   : ',JUL
    ?'             Février : ',FEV,'     Août      : ',AOU
    ?'             Mars    : ',MAR,'     Septembre : ',SEP
    ?'             Avril   : ',AVR,'     Octobre   : ',OCT
    ?'             Mai     : ',MAI,'     Novembre  : ',NOV
    ?'             Juin    : ',JUN,'     Décembre  : ',DEC
    ?'             *'
    ?' '
    *SET PRINT OFF
    @20,5 SAY 'Autre année (O/N) ? ' GET REP
    READ
  ENDIF
ENDDO

CLOSE ALL
RETURN
```

AM_STAT3.PRG

DEBUT DE BOUCLE

Lire l'année entrée par l'utilisateur

Vérifier s'il existe un enregistrement pour l'année souhaitée

S'il existe, totaliser les prêts mensuels de l'année considérée

Afficher et imprimer - le total
- le détail mensuel.

Proposer l'enchaînement sur une autre année.

FIN DE BOUCLE

AM_STAT4.PRG

```
*AM_STAT4.PRG

PRIVATE AN,ANNEE,REP,VTOT

CLEAR
SET COLOR TO W
@1,20 SAY '* Tableau par secteur/catégorie *'

SET COLOR TO BG
REP='0'
DO WHILE UPPER(REP)<>'N'

  ANNEE=SPACE(4)
  @4,5 SAY 'Année souhaitée : ' GET ANNEE PICTURE '####'
  READ
  AN=VAL(ANNEE)
  IF AN=0
    CLOSE ALL
    CLEAR
    RETURN
  ENDIF

  IF YEAR(DATE())=AN
    USE SECCAT
  ELSE
    AN=SUBSTR(ANNEE,3)
    USE ('SECCAT'+AN)
  ENDIF
  @4,5
  ACTIVATE WINDOW affstat
  SET PRINT ON
  ?'         Année : ',AN
  SCAN
    VTOT=CATA+CATB+CATC+CATD+CATE+CATF+CATG+CATH+CATI+CATJ+CATK+CATL+CATM+CATN+CATO+CATP+CATQ+CATR+CATS+CATT
    ?'         TOTAL secteur documentaire ',SECTEUR,' : 'VTOT
    ?' '
    ?'         Catégories de lecteurs : '
    ?'         A : ',CATA,'      K : ',CATK
    ?'         B : ',CATB,'      L : ',CATL
    ?'         C : ',CATC,'      M : ',CATM
    ?'         D : ',CATD,'      N : ',CATN
    ?'         E : ',CATE,'      O : ',CATO
    ?'         F : ',CATF,'      P : ',CATP
    ?'         G : ',CATG,'      Q : ',CATQ
    ?'         H : ',CATH,'      R : ',CATR
    ?'         I : ',CATI,'      S : ',CATS
    ?'         J : ',CATJ,'      T : ',CATT
    ?' '
    ?'         *'
    ?' '
  ENDSCAN
  CLOSE ALL
  SET PRINT OFF
  @20,3 SAY 'Autre tableau (O/N) ? ' GET REP
  READ
  DEACTIVATE WINDOW affstat
ENDDO

CLEAR
RETURN
```

Annotations manuscrites :

DEBUT DE BOUCLE

Lire l'année entrée par l'utilisateur

Ouvrir le fichier SECCAT*.dbf qui correspond à l'année demandée

Ouvrir une fenêtre (pour conserver l'intitulé)

Totaliser le nombre des prêts des catégories pour chaque secteur.

Afficher et imprimer pour chaque secteur
- le total par secteur
- le détail par catégories

Proposer d'enchaîner sur une autre année

Fermer la fenêtre

FIN DE BOUCLE

```
*AM_STAT5.PRG

PRIVATE NBDOC,NBPRET,NBLEC,NBACTIF

CLEAR
SET COLOR TO W
@1,24 SAY '* Etat documents et lecteurs *'

SET COLOR TO BG
USE DOCUMENT
NBDOC=RECCOUNT()
USE LECTEUR
NBLEC=RECCOUNT()
USE PRET
NBPRET=RECCOUNT()
USE PARAM
NBACTIF=ACTIFS
CLOSE ALL
ACTIVATE WINDOW affstat
SET PRINT ON
?'          Nombre total de documents      : ',NBDOC
?'          Nombre total de prêts en cours : ',NBPRET
?'          Nombre total de lecteurs       : ',NBLEC
USE CATEG
?'          CATEGORIE  Nombre     CATEGORIE  Nombre'
?' '
?'             A        ',CATA,'     K        ',CATK
?'             B        ',CATB,'     L        ',CATL
?'             C        ',CATC,'     M        ',CATM
?'             D        ',CATD,'     N        ',CATN
?'             E        ',CATE,'     O        ',CATO
?'             F        ',CATF,'     P        ',CATP
?'             G        ',CATG,'     Q        ',CATQ
?'             H        ',CATH,'     R        ',CATR
?'             I        ',CATI,'     S        ',CATS
?'             J        ',CATJ,'     T        ',CATT
?' '
CLOSE ALL
?'          Nombre de lecteurs actifs      : ',NBACTIF
?'          *'
?' '
SET PRINT OFF
WAIT '          Impression terminée...> '
DEACTIVATE WINDOW affstat

RETURN
```

AM_STAT5.PRG

Calculer
- le nombre de documents de la base
- le nombre de prêts "
- le nombre de lecteurs
- le nombre de lecteurs actifs

Afficher et imprimer
- ces données
ainsi que :
- le détail du nombre de lecteurs de chaque catégorie

Signaler la fin du tableau.

```
*AM_SAUVE.PRG

CLEAR

SET COLOR TO W
@1,30 SAY '* SAUVEGARDES *'
SET COLOR TO BG

RUN BACKUP c:\dbase\bibens a: /m
CLEAR
RETURN
```

AM_SAUVE.PRG

activer la commande DOS de Sauvegarde

```
*AM_SUPPR.PRG

PRIVATE REP

CLEAR
SET COLOR TO W
@1,29 SAY '* SUPPRESSIONS *'
SET COLOR TO BG

ACTIVATE WINDOW sitlec
@1,2 SAY '1 - Suppression document '
@3,2 SAY '2 - Suppression lecteur '
REP=0
@5,9 SAY 'Choix : ' GET REP PICTURE '#'
READ
DEACTIVATE WINDOW sitlec

DO CASE
  CASE REP=1
    DO AM_DOCSUPP
  CASE REP=2
    DO AM_LECSUPP
ENDCASE

CLEAR
RETURN
```

AM_SUPPR.PRG

Proposer les 2 types de suppression

Lire le choix

Cas 1
exécuter le sous-programme AM_DOCSUP

Cas 2
id. . . . . AM_LECSUP

```
*AM_DOCSUP.PRG

PRIVATE VINV,CONF

CLEAR
SET COLOR TO W
@1,23 SAY '* Suppression document *'
SET COLOR TO BG
```

Lire le n° inventaire

```
VINV=SPACE(8)
@4,5 SAY 'Entrez le N°inventaire : ' GET VINV PICTURE '#####X##'
READ
```

Vérifier si le document existe

```
USE DOCUMENT ORDER numinv
SEEK VINV
IF EOF()
  @20,5 SAY 'Document non enregistré...>'
  DO BEEP
  DO PAUSE
  CLOSE ALL
  CLEAR
  RETURN
ENDIF
```

Si oui,
initialiser les variables

```
VTYP=TYPDOC
VCOD=CODBARR
VCOT=COTE
VAN=ANNEENO
VTIT=TITRE
VM1=MOT1
VM2=MOT2
VM3=MOT3
VRES=RESER
VPR=PRET
```

afficher les coordonnées

```
@4,5
DO AM_AFFDOC
```

signaler si ce document est en prêt, attendre, quitter

```
IF VPR<>0
  @20,5 SAY 'Document en prêt, suppression impossible...>'
  DO BEEP
  DO PAUSE
  CLOSE ALL
  CLEAR
  RETURN
ENDIF
```

Demander confirmation de cette suppression

```
CONF='N'
@20,5 SAY 'Confirmez-vous cette suppression (O/N) ? ' GET CONF
READ
IF UPPER(CONF)='O'
```

Effacer l'enregistrement correspondant

```
  DELETE
  PACK
```

Effacer, si nécessaire, les réservations faites sur ce document, et le signaler, le cas échéant

```
  IF VRES<>0
    USE RESERVAT
    SCAN FOR CODBARR=VCOD
      DELETE
      PACK
    ENDSCAN
    @18,5 SAY 'Réservation(s) en cours supprimées.'
  ENDIF
  CLOSE ALL
```

Imprimer les coordonnées du document supprimé.

```
  SET CONSOLE OFF
  SET PRINT ON
  ?DATE(),'    Suppression document.'
  DO AM_AFFDOC
```

```
  SET PRINT OFF
  SET CONSOLE ON
ENDIF

CLOSE ALL
CLEAR
RETURN
```

```
*AM_LECSUPP.PRG

PUBLIC VCAT,VNOM,VPRE,VRES,VRUE,VCOP,VLIE,VNBP,VNBR,VMES,VVAL
PRIVATE REP

CLEAR
SET COLOR TO W
@1,26 SAY '* Suppression lecteur *'
SET COLOR TO BG

ACTIVATE WINDOW sitdoc
@1,2 SAY '1 - Accès par date de validité'
@3,2 SAY '2 - Accès par le code'
REP=0
@5,10 SAY 'Choix : ' GET REP PICTURE '#'
READ
DEACTIVATE WINDOW sitdoc


DO CASE
  CASE REP=1
    DO LECSUPD
  CASE REP=2
    DO LECSUPC
ENDCASE

CLEAR
RETURN


                              *PROCEDURES

Procedure LECSUPC

PRIVATE CONF

VCAR=0
@4,5 SAY 'Entrez le code du lecteur : ' GET VCAR PICTURE '###'
READ
USE LECTEUR ORDER numcar
SEEK VCAR
IF EOF()
  @20,5 SAY 'Lecteur non enregistré...>'
  DO BEEP
  DO PAUSE
  CLOSE ALL
  CLEAR
  RETURN
ENDIF
DO LECVAR
@4,5
DO AM_AFFLEC

IF VNBP<>0
  @20,5 SAY 'Prêt(s) en cours, suppression impossible...>'
  DO BEEP
  DO PAUSE
  CLOSE ALL
  CLEAR
  RETURN
ENDIF

CONF='N'
@20,5 SAY 'Confirmez-vous cette suppression (O/N) ? ' GET CONF
READ
IF UPPER(CONF)='O'
```

Déclarer les variables

Proposer 2 types de suppression de lecteur
Lire le choix

Cas 1
Exécuter la procédure LECSUPD

Cas 2
id . . . . LECSUPC

Lire le code du lecteur

Vérifier si ce lecteur existe

exécuter la procédure LECVAR (initialisation des variables)
afficher les coordonnées du lecteur correspondant

Indiquer si le lecteur a des prêts en cours, attendre, quitter

Demander confirmation de cette suppression, à l'utilisateur

```
  DELETE
  PACK
```

Effacer l'enregistrement du lecteur

```
  IF VNBR<>0
    USE RESERVAT
    SCAN FOR NUMCAR=VCAR
      DELETE
      PACK
    ENDSCAN
    @18,5 SAY 'Réservation(s) en cours supprimées.'
  ENDIF
```

Effacer ses réservations éventuelles

et le signaler, le cas échéant.

```
  CLOSE ALL
  CLEAR
  DO SUPCATEG

  SET CONSOLE OFF
  SET PRINT ON
  ?DATE(),'    Suppression lecteur.'
  DO AM_AFFLEC
  SET PRINT OFF
  SET CONSOLE ON
ENDIF

RETURN


Procedure LECSUPD

PRIVATE VDAT
```

Exécuter la procédure SUPCATEG

Imprimer les coordonnées du lecteur supprimé de la base

```
@4,15 SAY 'Cette suppression concernera les lecteurs : '
@5,15 SAY '  - qui n ont pas de prêt en cours'
@6,15 SAY '  - qui ne sont pas réinscrits depuis'
@7,15 SAY '    la date que vous indiquerez.'
VDAT={  /  /  }
@9,19 SAY 'Entrez cette date : ' GET VDAT
READ
IF VDAT={  /  /  }
  CLOSE ALL
  CLEAR
  RETURN
ENDIF
```

Expliquer le principe de ce type de supression

Lire la date entrée par l'utilisateur

```
@19,5 SAY 'Suppressions en cours...'
SET PRINT ON
?DATE(),'   Suppressions lecteurs non réinscrits depuis le : ',VDAT
```

Imprimer date et intitulé

DEBUT DE BOUCLE

```
SET CONSOLE OFF
USE LECTEUR IN 1 ORDER numcar
SCAN FOR VALID<=VDAT .AND. NBPRET=0
  VCAR=NUMCAR
  DO LECVAR
  DELETE
  PACK
  DO AM_AFFLEC
  IF VNBR<>0
    USE RESERVAT IN 2
    SELECT 2
    LOCATE FOR NUMCAR=VCAR
    DELETE
    PACK
    USE IN 2
  ENDIF
  DO SUPCATEG
ENDSCAN
SET CONSOLE ON
SET PRINT OFF
```

Rechercher chaque lecteur ayant atteint la limite de validité et n'ayant pas de prêt en cours

Initialiser les variables, effacer l'enrgt

Imprimer ses coordonnées

Effacer ses réservations éventuelles

Exécuter la procédure SUPCATEG.

FIN DE BOUCLE

```
CLOSE ALL
CLEAR
RETURN


Procedure LECVAR

VCAT=CATEG
VNOM=NOM
VPRE=PRENOM
VRES=RESID
VRUE=RUE
VCOP=CODPOST
VLIE=LIEU
VNBP=NBPRET
VNBR=NBRESER
VMES=MESSG
VVAL=VALID
RETURN
```

Initialiser les variables en leur affectant les valeurs des champs de l'enregistrement lecteur

```
Procedure SUPCATEG
USE CATEG IN 2
SELECT 2
DO CASE
  CASE VCAT='A'
    REPLACE CATA WITH CATA-1
  CASE VCAT='B'
    REPLACE CATB WITH CATB-1
  CASE VCAT='C'
    REPLACE CATC WITH CATC-1
  CASE VCAT='D'
    REPLACE CATD WITH CATD-1
  CASE VCAT='E'
    REPLACE CATE WITH CATE-1
  CASE VCAT='F'
    REPLACE CATG WITH CATG-1
  CASE VCAT='H'
    REPLACE CATH WITH CATH-1
  CASE VCAT='I'
    REPLACE CATI WITH CATI-1
  CASE VCAT='J'
    REPLACE CATJ WITH CATJ-1
  CASE VCAT='K'
    REPLACE CATK WITH CATK-1
  CASE VCAT='L'
    REPLACE CATL WITH CATL-1
  CASE VCAT='M'
    REPLACE CATM WITH CATM-1
  CASE VCAT='N'
    REPLACE CATN WITH CATN-1
  CASE VCAT='O'
    REPLACE CATO WITH CATO-1
  CASE VCAT='P'
    REPLACE CATP WITH CATP-1
  CASE VCAT='Q'
    REPLACE CATQ WITH CATQ-1
  CASE VCAT='R'
    REPLACE CATR WITH CATR-1
  CASE VCAT='S'
    REPLACE CATS WITH CATS-1
  CASE VCAT='T'
    REPLACE CATT WITH CATT-1
ENDCASE

USE IN 2
RETURN
```

Mettre à jour le total des lecteurs de la catégorie du lecteur supprimé de la base.

Pour ce faire, rechercher le compteur correspondant dans le fichier CATEG.

```
*AM_PARAMAJ.PRG

PRIVATE REP

CLEAR
SET COLOR TO W
@1,24 SAY '* PARAMETRES *'
SET COLOR TO BG
?CDOW(DATE())
??' '
??DATE()
?'Heure : '
??TIME()

REP=SPACE(1)
USE PARAM
DUR1 =DUREE1
DUR2 =DUREE2
PRMAX=MAXPRET
FERM =DATEFERM
OUV  =DATEOUV
INTREL=INTERREL
DURRES=DUREERES
CLOSE ALL

@6,21 SAY 'DUREE DU PRET TYPE 1 :'
@7,29 GET DUR1 PICTURE '##'
@8,21 SAY 'Validez ou modifiez.'
READ
DO WHILE DUR1 > 45
  @8,19 SAY 'Durée impossible, recommencez.'
  @7,29 GET DUR1 PICTURE '##'
  READ
ENDDO
@8,19

@10,21 SAY 'DUREE DU PRET TYPE 2 :'
@11,29 GET DUR2 PICTURE '##'
@12,21 SAY 'Validez ou modifiez.'
READ
DO WHILE DUR2 > 30
  @12,19 SAY 'Durée impossible, recommencez.'
  @11,29 GET DUR2 PICTURE '##'
  READ
ENDDO
@12,19

@14,20 SAY 'NOMBRE DE PRETS MAXIMUM :'
@15,30 GET PRMAX PICTURE '#'
@16,21 SAY 'Validez ou modifiez.'
READ
DO WHILE PRMAX < 3
@16,19 SAY 'Nombre impossible, recommencez.'
  @15,30 GET PRMAX PICTURE '#'
  READ
ENDDO
@16,19

@18,17 SAY 'DATES DE LA PROCHAINE FERMETURE : '
@20,22 GET FERM
@20,30 SAY ' - '
@20,33 GET OUV
@22,22 SAY 'Validez ou modifiez.'
READ
@22,22
@6,21 CLEAR
```

AM-PARAMAJ.PRG 1

Afficher le jour de la semaine
la date
l'heure

Initialiser les variables

Afficher la valeur
Lire la valeur entrée, si il y a eu modification
Vérifier si la modification est correcte

id

id

id
Sans contrôler la modification

```
@6,18 SAY 'INTERVALLE ENTRE DEUX RELANCES : '
@7,30 GET INTREL PICTURE '$'
@8,22 SAY 'Validez ou modifiez.'
READ
@8,22

@10,22 SAY 'DUREE DE RESERVATION : '
@11,30 GET DURRES PICTURE '$'
@12,22 SAY 'Validez ou modifiez.'
READ
@12,22

USE PARAM
REPLACE DUREE1 WITH DUR1
REPLACE DUREE2 WITH DUR2
REPLACE MAXPRET WITH PRMAX
REPLACE DATEFERM WITH FERM
REPLACE DATEOUV WITH OUV
REPLACE INTERREL WITH INTREL
REPLACE DUREERES WITH DURRES
CLOSE ALL

SET CONSOLE OFF
SET PRINT ON
?DATE(),'   PARAMETRAGE.'
?'Durée prêt type 1       : ',DUR1
?'Durée prêt type 2       : ',DUR2
?'Maximum de prêts        : ',PRMAX
?'Prochaine fermeture     : ',FERM,'-',OUV
?'Intervalle des relances: ',INTREL
?'Durée de réservation    : ',DURRES
?'*'
?' '
SET PRINT OFF
SET CONSOLE ON

CLEAR
RETURN
```

⋮
id
⋮
id

Remplacer les champs de l'enregistrement par les nouvelles valeurs

Imprimer les données de l'enregistrement modifié

```
*AM_MAJ.PRG

PRIVATE AN

USE DOCUMENT
SCAN
  REPLACE AN5 WITH AN4,AN4 WITH AN3,AN3 WITH AN2,AN2 WITH AN1,AN1 WITH 0
ENDSCAN
CLOSE ALL

AN=SUBSTR(STR(YEAR(DAT)),9)
COPY FILE SECCAT.DBF TO 'SECCAT'+AN+'.DBF'
USE SECCAT
SCAN
  REPLACE CATA WITH 0,CATB WITH 0,CATC WITH 0,CATD WITH 0,CATE WITH 0
  REPLACE CATF WITH 0,CATG WITH 0,CATH WITH 0,CATI WITH 0,CATJ WITH 0
  REPLACE CATK WITH 0,CATL WITH 0,CATM WITH 0,CATN WITH 0,CATO WITH 0
  REPLACE CATP WITH 0,CATQ WITH 0,CATR WITH 0,CATS WITH 0,CATT WITH 0
ENDSCAN

CLOSE ALL
RETURN
```

AM_MAJ.PRG

Décaler les valeurs des 5 champs compteurs annuels, dans chaque enregistrement document
AN5 disparait, AN1 redevient disponible pour la nouvelle année

- Copier le contenu du fichier SECCAT.DBF sur un nouveau fichier SECCATnn.DBF (nn = 2 derniers chiffres de l'année écoulée)
- Mettre les compteurs à zéro dans le fichier SECCAT.DBF, pour chaque enregistrement

AM_FINRES.PRG

```
*AM_FINRESER.PRG

CLEAR
SET CONSOLE OFF
SET PRINT ON
?DATE(),'  Réservations périmées.'
?'  Code lecteur  Code document   Date limite de réser.'
USE RESERVAT
SCAN FOR DATELIM<DATE()
  ?'      ',NUMCAR,'     ',CODBARR,'        ',DATELIM
  DELETE
  PACK
ENDSCAN
CLOSE ALL
?' '
SET PRINT OFF
SET CONSOLE ON

RETURN
```

Localiser les réservations périmées
Imprimer la liste
Les effacer du fichier RESERVAT

```
*AM_PENAL.PRG

PRIVATE VCAR,VPEN

USE PRET
SCAN FOR DATERET<DATE()
  VCAR=NUMCAR
  VRET=DATERET
  USE LECTEUR IN 2 ORDER numcar
  IF MONTH(DATE())=MONTH(VRET)
    VPEN=DATE()+(DAY(DATE())-DAY(VRET))
  ELSE
    VPEN=DATE()+((31-DAY(VRET))+DAY(DATE()))
  ENDIF
  SELECT 2
  SEEK VCAR
  REPLACE PENAL WITH VPEN
  USE IN 2
ENDSCAN
CLOSE ALL

RETURN
```

AM-PENAL.PRG

DEBUT DE BOUCLE

Rechercher les prêts dont la date retour est dépassée

Initialiser les variables

Retrouver pour chacun de ces prêts, le lecteur

Calculer sa date de pénalisation

Remplacer le champ 'pénalisation' par cette date

FIN DE BOUCLE

```
*AM_AFFLEC.PRG

?'   Code            : ',VCAR
?'   Catégorie       : ',VCAT
?'   Nom             : ',VNOM
?'   Prénom          : ',VPRE
?'   Adresse         : ',VRES
?'                     ',VRUE
?'                     ',VCOP,' ',VLIE
?'   Prêt(s)         : ',VNBP
?'   Réservation(s) : ',VNBR
?'   Observation     : ',VMES
?'   Validité        : ',VVAL
?'   *'
?' '
RETURN
```

AM_AFFLEC.PRG

Afficher le contenu des champs d'un enregistrement LECTEUR

```
*AM_AFFDOC.PRG

?'   Code          : ',VCOD
?'   Type          : ',VTYP
?'   Cote          : ',VCOT
?'   N°inventaire  : ',VINV
?'   Année,N°      : ',VAN
?'   Titre         : ',VTIT
?'   1er mot-titre : ',VM1
?'   2e  mot-titre : ',VM2
?'   3e  mot-titre : ',VM3
?'   Réservation   : ',VRES
?'   Prêt          : ',VPR
?'   *'
?' '
RETURN
```

AM_AFFDOC.PRG

Afficher le contenu des champs d'un enregistrement DOCUMENT

